

Fachhochschule Mittweida (FH)

University of Applied Sciences

Fachbereich Informationstechnik & Elektrotechnik

Studiengang Multimediatechnik

Diplomarbeit

**Neukonzeption und Umsetzung der Internetpräsenz und des Intranets für die
Klinikum Schönebeck GmbH auf Basis des Content Management Systems**

TYPO3

Begin der Arbeit: 01. Mai 2009

Abgabe der Arbeit: 21. August 2009

Betreuender Hochschullehrer (gleichzeitig Erstprüfer):
Prof. Dr.-Ing. Frank Zimmer

Zweitprüfer:
Dipl. Betriebswirt (FH) Thomas Beyer

Vorgelegt von:
Torsten Zichner

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Chemnitz, den 20. August 2009

Torsten Zichner

Vorwort

Diese vorliegende Diplomarbeit ist der Abschluss meines Studiums der Multimediatechnik an der Hochschule Mittweida (FH). An dieser Stelle möchte ich mich besonders bei Ina Heydecke bedanken für Ihre Unterstützung in allen Belangen des privaten Lebens und dem damit gegebenen Rückhalt.

Auf Seiten der Hochschule gilt mein besonderer Dank Herrn Prof. Dr.-Ing. Frank Zimmer für seine Geduld und Unterstützung.

Abschließend möchte ich mich noch bei meinem Betreuer Herrn Thomas Beyer bedanken, für die immer freundschaftliche und lockere Arbeitsatmosphäre und Denkanstöße, wenn sie von Nöten waren.

Torsten Zichner

Kurzfassung

Im hier vorliegenden Fall wird für die Klinikum Schönebeck GmbH eine neue Internetpräsenz geschaffen, welche das Unternehmen nach Außen hin repräsentieren soll und Informationen für Interessierte zur Verfügung stellt. Weiterhin wird ein Intranet erstellt, welches nur für den Informationsaustausch der Mitarbeiter und Ärzte zur Verfügung steht, um die Kommunikation innerhalb des Unternehmens zu fördern und zu verbessern.

Der Internetauftritt wie auch das Intranet sollen hierbei mittels des Content Management Systems TYPO3 erschaffen und verwaltet werden.

Stichworte

Internet, Internetseite, Intranet, Content Management, Content Management System (CMS), TYPO3

Abstract

In this work a website for the Klinikum Schönebeck GmbH is created, which shall represent the company and provide information for everyone who's interested. Furthermore an intranet is created, which is designed for information exchange between employees and doctors, to advance and enhance the communication within the company.

The website as well as the intranet will be created and managed by the content management system (cms) TYPO3.

Keywords

internet, website, intranet, content management, content management system (cms), TYPO3

Inhaltsverzeichnis

Vorwort	V
Kurzfassung	VII
Abstract	IX
Inhaltsverzeichnis	XI
Abkürzungsverzeichnis	XV
Abbildungsverzeichnis	XVII
1. Einleitung	1
1.1. Zielsetzung.....	1
1.2. Aufbau der Diplomarbeit	2
1.3. Abgrenzung.....	2
2. Grundlagen	3
2.1. Begriffserklärungen	3
2.1.1. Content Management System	3
2.1.2. Intranet	4
2.1.3. weitere Begriffe.....	5
2.2. TYPO3	6
2.2.1. Vorstellung	6
2.2.2. Vor und Nachteile von Open-Source-Systemen	7
2.2.3. Mindestsystemvoraussetzungen	8
2.2.4. Funktionalität	9
2.2.4.1. TYPO3- und HTML Templates.....	9
2.2.4.2. TypoScript.....	10
2.2.5. Oberfläche	11
2.3. Situationsanalyse	12
2.3.1. Auftraggeber / Auftragnehmer	12
2.3.2. Aktuelle Online-Präsentation Klinikum Schönebeck GmbH.....	13
2.4. Pflichtenheft	13
2.4.1. Relevante Punkte	14
2.4.1.1. TYPO3 Basisentwicklung.....	14
2.4.1.2. TYPO3 Extensions.....	15

2.4.1.3. Seiten- und Rechtestruktur des Intranet	17
3. Umsetzung	19
3.1. System aufsetzen.....	19
3.2. Einrichten des Systems.....	20
3.2.1. Benutzereinstellungen	20
3.2.2. Extensions	20
3.3. TYPO3- und HTML-Templates	22
3.3.1. HTML-Templates.....	23
3.3.1.1. Webpräsenz.....	23
3.3.1.2. Intranet.....	25
3.3.2. TYPO3 Template.....	25
3.4. Grundstruktur und Seitenaufbau	29
3.4.1. Anlegen der Ordner- und Seitenhierarchie	29
3.4.1.1. Webpräsenz.....	30
3.4.1.2. Intranet.....	32
3.5. Module (Frontend)	32
3.5.1. Terminkalender.....	33
3.5.1.1. Anforderungen	33
3.5.1.2. Variantenanalyse	33
3.5.1.3. Entscheidung	35
3.5.1.4. Konfiguration.....	36
3.5.2. Volltextsuche	40
3.5.2.1. Anforderungen	40
3.5.2.2. Variantenanalyse	40
3.5.2.3. Entscheidung	42
3.5.2.4. Konfiguration.....	42
3.5.3. News	44
3.5.3.1. Anforderungen	44
3.5.3.2. Variantenanalyse	45
3.5.3.3. Konfiguration.....	45
3.5.4. Weitere Extensions.....	45
3.5.4.1. Flash Page Header (flash_pageheader)	46
3.5.4.2. Powermail (powermail)	47
3.5.4.3. Tip-A-Friend Plus (tipafriend_plus)	48

3.6. Closed User Groups / Rechtevergabe	48
3.6.1. Gruppen	49
3.6.2. Webpräsenz	49
3.6.3. Intranet	53
3.6.3.1. Rechte - Backend	54
3.6.3.1.1. Zielsetzung	54
3.6.3.1.2. Umsetzung	54
3.6.3.1.2.1. Rechte	55
3.6.3.1.2.2. Einstiegspunkt	55
3.6.3.1.2.3. Besitz	56
3.6.3.1.2.4. User	58
3.6.3.2. Rechte - Frontend	58
3.6.3.2.1. Zielsetzung	59
3.6.3.2.2. Umsetzung	59
3.7. Datensicherheit	60
3.7.1. Schutz vor Datenverlust	60
3.7.2. Schutz vor Fremdzugriff und Diebstahl oder Manipulation	60
4. Zusammenfassung und Ergebnisse	63
5. Ausblick.....	65
6. Anhang	i
6.1. HTML-Templates	i
6.2. TypoScript.....	iv
7. Quellenverzeichnis	xi

Abkürzungsverzeichnis

BE	Backend
CD	Corporate Design
CI	Corporate Identity
CMS	Content Management System
FE	Frontend
FTP	File Transfer Protocol
GNU	GNU is Not Unix
GPL	GNU General Public License
HTTP	Hypertext Transfer Protocol
MySQL	My ¹ Structured Query Language (Strukturierte Abfrage Sprache)
o.V.	ohne Verfasser
PHP²	PHP : H ypertext P reprocessor (ursprünglich: P ersonal H ome P age Tools)
PSD	Photoshop Document
TCP/IP	Transmission Control Protocol / Internet Protocol
WCMS	Web Content Management System
W3C	World Wide Web Consortium

¹ „Die Herkunft des Namens *MySQL* kann heute nicht mehr genau rekonstruiert werden. Seit 1996 wurden diverse Bibliotheken und Tools mit dem Präfix *My* geschrieben. Es wird spekuliert, dass der Name *My* der Tochter des Mitbegründers Michael Widenius vielleicht auch der Ursprung des Namens *MySQL* sein könnte“ - <http://de.wikipedia.org/wiki/MySQL> - Stand 03. August 2009

² o.V. - PHP - <http://de.wikipedia.org/wiki/Php> - Stand 03. August 2009

Abbildungsverzeichnis

Abb. 1 - Oberfläche des Backend des WCMS TYPO3	12
Abb. 2 - Grundstruktur der Webpräsenz	23
Abb. 3 - Grundstruktur des Intranets	25
Abb. 4 - die erste Seite in einem neuen TYPO3 Projekt	26
Abb. 5 - eine „Wurzel“-Seite mit dem TYPO3-Template	26
Abb. 6 - erste Ebene der Baumstruktur der Webpräsenz.....	30
Abb. 7 - Auszug aus Seitenbaum - Service und Datenmanagementordner aufgeklappt....	31
Abb. 8 - Seitenbaum des Intranets - geschützt	32
Abb. 9 - Auszug aus dem Seitenbaum des Intranet, mit den Seiten für Termine	36
Abb. 10 - Auszug aus der Sysordnerstruktur des Intranet, mit den Ordnern für Termine....	36

1. Einleitung

Das Internet ist heutzutage eines der wichtigsten Medien und kann vielfältig und gewinnbringend für ein Unternehmen eingesetzt werden. Gepaart mit geringen laufenden Kosten ist es aus dem Alltag nicht mehr wegzudenken.

Im Falle der Klinikum Schönebeck GmbH soll es einerseits zur Verbreitung und Veranschaulichung von Informationen sowie zur Werbung und Repräsentation nach Außen eingesetzt werden und andererseits dem Austausch von geschützten Informationen mehrerer Parteien innerhalb des Unternehmens dienen.

Verbunden mit der großen Skalierbarkeit des WCMS TYPO3³ für beide Anwendungsgebiete ist es möglich, eine einheitliche Benutzeroberfläche zu gewährleisten, um die Verwaltung beider Präsenzen so einfach wie möglich zu halten.

1.1. Zielsetzung

Ziel dieser Arbeit ist es einerseits die bestehende Internetpräsenz der Klinikum Schönebeck GmbH komplett neu zu gestalten. Wert gelegt wird hierbei auf Barrierefreiheit (beziehungsweise Barrierearmut) und damit einhergehend eine saubere Trennung von Design und Inhalt. Durch Verwendung des CMS TYPO3 soll es dem Kunden ermöglicht werden, die Pflege der Seiten und Inhalte selbst zu übernehmen, um Informationen zeitnah und mit wenig Aufwand veröffentlichen zu können.

Weiterhin soll für die Klinikum Schönebeck GmbH ein Intranet aufgebaut werden, welches die Kommunikation innerhalb des Unternehmens erleichtert und optimiert. Es soll eine Plattform geschaffen werden, in der die Ärzte und Mitarbeiter einfach und ohne Probleme Zugang zu bestimmten Informationen und Daten erhalten. Dabei soll ebenfalls das CMS TYPO3 verwendet werden, welches nach Fertigstellung des

³ o.V. - TYPO3 - <http://de.wikipedia.org/wiki/TYPO3> - Stand 06. August 2009

Intranets der Klinikum Schönebeck GmbH eine eigenständige Verwaltung aller Informationen und Nutzer ermöglicht.

1.2. Aufbau der Diplomarbeit

Die folgende Arbeit beginnt mit der Erläuterung einiger Grundlagen, welche zum Verständnis wichtig sind. Hierzu gehören Erklärungen zu wichtigen Begriffen wie CMS und Intranet, gefolgt von einer Einführung in TYPO3.

Anschließend wird sich mit der Umsetzung des Themas beschäftigt. Der Aufbau ist hierbei an der chronologisch korrekten Vorgehensweise orientiert und beginnt mit der Aufsetzung der Server. Über die Grundkonfiguration des CMS, das Einrichten der benötigten Erweiterungen und der abschließenden Erstellung der nötigen Rechtestruktur findet die Arbeit ihren Abschluss.

Am Ende erfolgt eine Zusammenfassung der Ergebnisse und es wird ein Ausblick auf die Zukunft dieser Arbeit geben.

1.3. Abgrenzung

Der Inhalt dieser Arbeit befasst sich mit dem Erschaffen der Internetpräsenz, dem Intranet und dem Anpassen des CMS TYPO3 für den täglichen Gebrauch der Klinikum Schönebeck GmbH.

Es wird keine Konzeption über den möglichen Aufbau oder das Design angefertigt. Beides steht als Ausgangsmaterial zur Verfügung.

Beide Anwendungen (die Internetpräsenz und das Intranet) werden auf einem Webserver des Anbieters Mittwald CM Service GmbH & Co KG⁴ mit vorinstalliertem TYPO3 eingerichtet.

⁴ o.V. - Mittwald - <http://www.mittwald.de> - Stand 05. August 2009

2. Grundlagen

Nachfolgend werden die für diese Arbeit benötigten Grundlagen näher gebracht. Anmerkend sei hier jedoch erwähnt, dass zum Verständnis der Arbeit der Leser mit den zugrunde liegenden Technologien wie HTML und CSS bereits etwas vertraut sein sollte.

In Kapitel 2.1 erfolgt eine Begriffserklärung der weiterführenden für diese Arbeit wichtigen Begriffe. Alle spielen in dieser Arbeit eine grundlegende Rolle und sollen im Sinne der Arbeit erläutert werden.

Weiterhin wird in Kapitel 2.3 auf die derzeitige Situation eingegangen. Dies beinhaltet eine Kurzvorstellung des Auftraggebers – der Klinikum Schönebeck GmbH – und eine Analyse der alten Internetpräsenz.

Abschließend wird auf das Pflichtenheft eingegangen, welches die Grundlage der kompletten Diplomarbeit ist.

2.1. Begriffserklärungen

Begriffe, welche häufig Erwähnung in dieser Arbeit finden oder für das Verständnis besonders wichtig sind, sollen hier näher erläutert werden.

2.1.1. Content Management System

Je umfangreicher Webseiten sind, deren Inhalt stets aktuell gehalten werden soll, umso größer wird der Aufwand, dies zu bewerkstelligen. Der Zeitaufwand steigt enorm und vor Allem auch die Kosten, wenn die benötigten Aktualisierungen nur von Programmierern oder Personen mit den nötigen Kenntnissen der verwendeten Programmiersprachen durchgeführt werden können.

In diesem Fall sollte man auf ein Content Management System zurückgreifen, welches durch nachfolgende Formulierung aus der online verfügbaren Enzyklopädie Wikipedia näher erläutert wird:

„Ein Content-Management-System (übersetzt: Inhaltsverwaltungssystem) ist ein System, das die gemeinschaftliche Erstellung und Bearbeitung von Inhalt, bestehend aus Text- und Multimedia-Dokumenten, ermöglicht und organisiert, meist für das World Wide Web. Ein Autor kann ein solches System auch ohne Programmier- oder HTML-Kenntnisse bedienen. Der darzustellende Informationsgehalt wird in diesem Zusammenhang als Content (Inhalt) bezeichnet“⁵

CMS, welche wie im Fall dieser Arbeit ausschließlich im Web benutzt werden, werden auch Web Content Management Systeme (WCMS) genannt. Sie bieten einfache Möglichkeiten zur Verwaltung und Pflege von Inhalten auf Internetseiten und sind somit auch von Personen ohne Kenntnisse von Programmiersprachen leicht zu verwenden.

2.1.2. Intranet

„Ein Intranet, von lateinisch intra für innen und englisch net für Netz, ist ein Rechnernetz, das im Gegensatz zum Internet nicht öffentlich ist.“⁶

Es findet somit Verwendung in einem unternehmensinternen Rechnernetzwerk, welches wie das Internet selbst auf den Protokollen TCP/IP und HTTP beruht und somit über die grafische Benutzeroberfläche von Webbrowsern benutzt werden kann.

Auch die Informationen werden wie im Internet von einem Webserver bereitgestellt. Die angebotenen Daten werden in einer an den Webserver angebundenen Datenbank gespeichert.

Die Ziele für den Einsatz des Intranets sind⁷:

⁵ o.V. - Content-Management-System - <http://de.wikipedia.org/wiki/Content-Management-System> - Stand 03. August 2009

⁶ o.V. - Intranet - <http://de.wikipedia.org/wiki/Intranet> - Stand 03. August 2009

⁷ o.V. - Intranet - <http://de.wikipedia.org/wiki/Intranet> - Stand 03. August 2009

- das Beschleunigen von innerbetrieblichen Informationsströmen
- das Nutzen einer gemeinsamen Datenbank, um den Zugriff auf Informationen für alle Personen zu vereinfachen
- die Möglichkeit, den Zugriff und die Nutzung einzelne Informationen auszuwerten

2.1.3. weitere Begriffe

Innerhalb dieser Arbeit finden weitere Begriffe Verwendung die hier kurz erklärt werden.,

Corporate Identity

„Corporate Identity (auch Unternehmensidentität) ist der abgestimmte Einsatz von Verhalten, Kommunikation und Erscheinungsbild nach innen und außen. Basis dafür ist das Unternehmensleitbild, welches durch die Corporate Identity mit Leben gefüllt wird. Ziel der Corporate Identity ist eine nachhaltige Unternehmensentwicklung.“⁸

Im Fall dieser Diplomarbeit bezieht sich die Abkürzung CI überwiegend auf einen Teilbereich der Corporate Identity: das Corporate Design (CD).

Corporate Design (CD)

„Unter Corporate Design wird die visuelle Identität verstanden, die als Erscheinungsbild Teil der Corporate Identity ist. Corporate Design findet Anwendung bei Gestaltung von Zeichen, Arbeitskleidung, Formularen, Architektur der Betriebsgebäude, Farbgebung und so weiter...“⁹

⁸ o.V. - Corporate Identity - http://de.wikipedia.org/wiki/Corporate_Identity - Stand 05. August 2009

⁹ o.V. - Corporate Identity - http://de.wikipedia.org/wiki/Corporate_Identity - Stand 05. August 2009

Barrierefreiheit

„Barrierefreiheit bedeutet, dass Gegenstände, Medien und Einrichtungen so gestaltet werden, dass sie von jedem Menschen unabhängig von einer eventuell vorhandenen Behinderung uneingeschränkt benutzt werden können.“¹⁰

Um diese Barrierefreiheit im Internet zu erreichen, ist auf die Einhaltung von Webstandards zu achten. Ebenso muss hierfür eine Trennung von Inhalt und Design vorliegen. Dies wird durch den korrekten Einsatz von Cascading Style Sheets (CSS) erreicht.¹¹

Da vollständige Barrierefreiheit selten, beziehungsweise durch den Einsatz verschiedener Gestaltungsmittel wie Flashelemente und ähnlichem nie erreicht werden kann, wird in diesem Zusammenhang auch häufig von Barrierearmut gesprochen.

2.2. TYPO3

TYPO3 ist das in dieser Arbeit verwendete WCMS. Es dient sowohl der Entwicklung als auch der Pflege der Webseite. Die Nutzung dieses Systems ist durch das Pflichtenheft vorgegeben.

2.2.1. Vorstellung

TYPO3 ist ein WCMS und wurde von Kasper Skårhøj seit 1997 entwickelt. 3 Jahre später wurde TYPO3 als Open-Source-Software unter der GPL¹² von Skårhøj freigegeben und seit 2002 in der Version 3.0 von einer größer werdenden Entwickler-Community betreut und voran gebracht. TYPO3 4.0 ist seit 2006 veröffentlicht und brachte eine Neuauflage der Nutzeroberfläche, das Arbeiten mit „Workspaces“¹³ und

¹⁰ o.V. - Barrierefreiheit - <http://de.wikipedia.org/wiki/Barrierefreiheit> - Stand 05. August 2009

¹¹ o.V. - Barrierefreies Internet - http://de.wikipedia.org/wiki/Barrierefreies_Internet - Stand 05. August 2009

¹² GPL ist eine Lizenz welche das Projekt zur freien Nutzung für andere bereitstellt. Dieses kann hierbei auch kommerziell genutzt werden. Jedoch dürfen Projekte, die auf diesen Projekt aufbauen, ebenfalls nur unter der GPL lizenziert werden und sind somit ebenfalls für jedermann einseh- und nutzbar. - o.V. GPL - <http://de.wikipedia.org/wiki/Gpl> - Stand 07. August 2009

¹³ Unterscheidung von Inhalten in einer so genannten veröffentlichten Ansicht und einer Entwurfsansicht.

ein integrierter Database Abstraction Layer (DBAL), welcher die Nutzung von unterschiedlichen Datenbanksystemen gewährleistet.

2.2.2. Vor und Nachteile von Open-Source-Systemen

Die Zugehörigkeit zu den Open Source CMS und dem unterliegen der GPL resultiert in einer völligen Kostenfreiheit bei Anschaffung und Verwendung von TYPO3.

Daraus ergeben sich Vorteile wie aber auch Nachteile. Der Vorteil von Open Source liegt auf der Hand. Das Einsparpotenzial gegenüber lizenzpflichtigen WCMS ist ein sehr häufiges Entscheidungskriterium bei der Wahl des Systems. Auch Erweiterungen und deren Nutzung sind im Regelfall kostenfrei und werden vielfältig angeboten. Ein Nachteil der Open Source Systeme ist in diesem Fall der wesentlich höhere erforderliche Kenntnisstand. Installation und Wartung müssen selbst durchgeführt werden und auch die Konfiguration des Systems kann ohne Support durch den Hersteller aufwendiger werden. Mehrkosten in Form von erhöhtem Stundeneinsatz können die Folge sein.

Im Gegenzug dazu werden kostenpflichtige Systeme im Allgemeinen bereits fertig installiert und konfiguriert. Um Installation und technische Wartung wie zum Beispiel Updates muss man sich nicht kümmern. Es geht auch hier nicht komplett ohne weiterführende Kenntnisse, und administrative Aktionen wie Rechtevergabe oder Backups müssen eventuell selbständig durchgeführt werden. Aber der Aufwand in diesem Bereich fällt wesentlich geringer aus und bei Problemen steht meist eine Herstellerhotline oder Support in anderer Art und Weise zur Verfügung.

Der Mangel an Support durch eine Firma kann bei vielen WCMS jedoch zum Teil durch die Leistungen der Community, welche das CMS unterstützt, wieder gut gemacht werden. Zahlreiche Webseiten und öffentliche Foren mit weitergehenden Informationen bieten hier Hilfe an oder liefern Lösungsvorschläge für Probleme.

2.2.3. Mindestsystemvoraussetzungen

Die Mindestsystemvoraussetzungen werden durch das verwendete WCMS bestimmt. In dieser Arbeit wurde TYPO3 in der Version 4.2.1 verwendet. Daraus resultieren folgende Mindestsystemanforderungen, die für den reibungslosen Betrieb erbracht werden müssen.¹⁴

Als Betriebssystem auf dem Server sind prinzipiell Unix (eg Linux), Windows oder Mac möglich und als Webserver sowohl Apache als auch IIS.

Alle TYPO3 Versionen ab 4.2.0 verlangen als serverseitige Skriptsprache PHP ab der Version 5.2. TYPO3 Versionen vor 4.2.0 können sowohl mit PHP 4 als auch mit PHP 5 betrieben werden. In dieser Arbeit läuft auf dem Server die PHP Version 5.2.5.

Als Datenbank wird MySQL oder ein anderes Datenbanksystem wie zum Beispiel Oracle benötigt. In dieser Arbeit wird auf MySQL in der Version 5.0 zurückgegriffen.

Weitere optionale jedoch empfohlene Voraussetzungen sind:

GDlib (seit PHP 4.3 in PHP integriert)¹⁵, Freetype sowie GraphicsMagick oder ImageMagick.

- GDlib ist eine Grafikbibliothek. Hierdurch kann man verschiedene Grafiken wie Linien und Kreise sowie Ausschnitte aus Bildern miteinander kombinieren.
- Freetype wird in Verbindung mit GDlib benötigt, um Schriften in Bilder zu rendern. So ist es in TYPO3 zum Beispiel möglich, grafische Menüs zu erstellen, indem man die Navigationsstruktur in die Hintergrundbilder einarbeitet.
- Mit Hilfe von ImageMagick oder GraphicsMagick ist möglich, Grafiken zu bearbeiten - zum Beispiel Skalieren oder Rotieren - und in unterschiedliche Formate zu konvertieren

¹⁴ o.V. - System Requirements - <http://TYPO3.org/1275.0.html> - Stand 04. August 2009

¹⁵ vgl. Andreas Stöckl, Frank Bongers (2006): Einstieg in TYPO3 4.0; 2., aktualisierte und erweiterte Auflage 2006, Seite 30

In dieser Arbeit wird das TYPO3 System auf einem Server des Providers Mittwald CM Service GmbH & Co KG¹⁶ gehostet. Dieser Hoster bietet spezielle Server mit der Möglichkeit der Vorinstallation einiger bekannter WCMS an, darunter auch TYPO3.

Der verwendete Server erfüllt somit alle Anforderungen, die von TYPO3 gestellt werden und das WCMS ist bereits für diesen Server grundkonfiguriert. Die technische Wartung entfällt somit. Das Open-Source-System ist somit zwar weiterhin kostenfrei, die verwendete Serverstruktur jedoch nicht.

2.2.4. Funktionalität

TYPO3 ist eine plattformunabhängige Serverapplikation (Kapitel 2.2.3), für deren Bedienung ein Webbrowser ausreichend ist. Es bietet sehr viele Entwicklungs- und Erweiterungsmöglichkeiten.

Eine Administrationsoberfläche, hier Backend (BE) genannt, steht sowohl Administratoren wie auch Redakteuren zur Verfügung. Hier werden Einstellungen von Seitenstruktur (Seitenbaum) bis hin zu Berechtigungen und Zugriffsmöglichkeiten gesteuert und die Seiten und Inhalte verwaltet.

Die nach Außen hin sichtbare Webpräsenz wird Frontend (FE) genannt.

2.2.4.1. TYPO3- und HTML Templates

Die Trennung von Design und Inhalt ist grundlegender Bestandteil von TYPO3. Dem Redakteur wird über das Backend ermöglicht, Inhalte und Informationen in Datensätzen zu hinterlegen. Dies tut er in einer hierarchischen Struktur, welche den Seitenbaum der Seite im Frontend widerspiegelt. Wie und wo auf einer Seite diese Informationen jedoch im Frontend dargestellt werden sollen, muss zuvor definiert werden.

¹⁶ o.V. - Mittwald - <http://www.mittwald.de> - Stand 05. August 2009

Hierfür benötigt man ein Template, welches in Summe alle Angaben umfasst, damit TYPO3 weiß, welcher Datensatz wo und wie im Frontend dargestellt werden soll. Der Begriff des Templates ist hier jedoch etwas umfangreicher zu verstehen als in anderen Content Management Systemen oder in verschiedenen HTML-Editoren. Das Template im Sinne von TYPO3 ist keine HTML-Datei mit Markern¹⁷ oder anderen Steuerelementen, sondern eine Ansammlung von Vereinbarungen. Dieses Konzept des HTML-Templates kann in TYPO3 jedoch auch verwendet werden, und wird überwiegend „HTML Designvorlage“ genannt - an manchen Stellen innerhalb TYPO3's wird jedoch auch der Begriff „Template“ für die Designvorlagen verwendet. Um Verwirrungen beider vorzubeugen, werden nachfolgend die Begriffe „TYPO3-Template“ und „HTML-Template“ (beziehungsweise „HTML-Designvorlage“) verwendet.

Das TYPO3-Template umfasst eine Abfolge von Anweisungen, wie eine Seite zu formatieren ist, wo die Inhalte dargestellt werden sollen, wie Menüs aufgebaut sein sollen und vieles mehr. Diese Anweisungen sind hierbei in einer TYPO3-eigenen Sprache verfasst, die als TypoScript bekannt ist.¹⁸

2.2.4.2. TypoScript

Obwohl der Name TypoScript an Sprachen wie JavaScript oder VBScript erinnert, so ist es jedoch keine Programmiersprache. Die Aufgabe von TypoScript besteht vielmehr in der „baumartig hierarchisch geordneten Ablage von Informationen“¹⁹, die nachfolgend vom WCMS für Darstellungs- und Konfigurationszwecke genutzt wird.

Einige Konzepte die Programmiersprachen enthalten, werden aber auch von TypoScript genutzt. So ist es möglich, den eigentlichen Code zu kommentieren, Leerräume werden ignoriert und Informationen können nach Datentyp unterschieden werden. Auch sind Kontrollstrukturen wie Entweder-Oder (if else) möglich, jedoch keine Schleifen. In Summe ist ein Quelltext von TypoScript eine Zusammenfassung

¹⁷ Ein Platzhalter innerhalb einer HTML-Seite, welcher durch Verwendung eines WCMS mit darzustellendem Inhalt ersetzt wird.

¹⁸ vgl. Andreas Stöckl, Frank Bongers (2006): Einstieg in TYPO3 4.0; 2., aktualisierte und erweiterte Auflage 2006, Seite 106

¹⁹ vgl. Andreas Stöckl, Frank Bongers (2006): Einstieg in TYPO3 4.0; 2., aktualisierte und erweiterte Auflage 2006, Seite 365

von Deklarationen von Objektinstanzen, deren Eigenschaften Werte zugewiesen werden.²⁰

Eingesetzt wird TypoScript an drei verschiedenen Stellen. Diese sind die „TypoScript Templates“, das „Page TSconfig“ und das „User TSconfig“.

Ersteres wird zur Steuerung des Seitenaufbaus verwendet, der den Bauplan für die Seiten innerhalb des Seitenbaums einer TYPO3-Webseite darstellen (Kapitel 2.2.4.1).

Das „Page TSconfig“ findet innerhalb der Definition eines jeden Datensatzes (page record) einer TYPO3 Seite Verwendung. Hier gibt es für jede Seite ein eigenes Eingabefeld, über das individuell für die einzelnen Seiten TypoScript-Anweisungen eingefügt werden können. Es ist auch möglich, die Gültigkeit dieser Anweisungen für Kindelemente²¹ zu bestimmen.

Das „User TSconfig“ ist dazu da, die Eigenschaften einzelner User oder auch ganzer Usergruppen mit zu bestimmen und zu beeinflussen.²²

2.2.5. Oberfläche

Die Oberfläche von TYPO3 ist schlicht und einfach gehalten. Sie basiert auf der Verwendung von Frames und ist im Spaltendesign aufgebaut. Überwiegend wird ein dreispaltiger Aufbau genutzt, welcher sich aufteilt in eine Modulleiste, eine Navigationsleiste und in eine Detailansicht.

- Die Modulleiste ist hierbei die oberste Schicht und bietet Module an, in welchen der Administrator und die Redakteure arbeiten können. Die zur Auswahl stehenden Module können über die Vergabe der Rechte bestimmt werden. So ist es möglich, die Anzahl der Module für Redakteure auf die Teile zu beschränken, welche Sie für Ihre Arbeit benötigen.

²⁰ vgl. Andreas Stöckl, Frank Bongers (2006): Einstieg in TYPO3 4.0; 2., aktualisierte und erweiterte Auflage 2006, Seite 366

²¹ Seiten welche dem aktuellen Element innerhalb der Baumstruktur untergeordnet sind.

²² vgl. Andreas Stöckl, Frank Bongers (2006): Einstieg in TYPO3 4.0; 2., aktualisierte und erweiterte Auflage 2006, Seite 365 und 366

- Die zweite Spalte ist abhängig vom gewählten Menüpunkt und bietet weitergehende Informationen an. In den meisten Modulen enthält sie eine Navigationsleiste, welche den Aufbau der Seite, wie sie von den Administratoren und Redakteuren angelegt wurde, widerspiegelt. Sie ist „auf- und zuklappbar“, um Platz sparend verwendet werden zu können.
- In der dritten und letzten Spalte erfolgt in der Regel die Bearbeitung der einzelnen Datensätze. Sie stellt somit den eigentlichen Arbeitsbereich vor. Hier werden Details zu den in der Navigation gewählten Seiten dargestellt und Datensätze eingefügt, angepasst oder gelöscht.

Seit TYPO3 4.2 gibt es einige Anpassungen an der Backendoberfläche. Sie basiert nicht länger auf Framesets, sondern es kommt ein iFrame²³ zum Einsatz. Weiterhin ist eine neue Kopfleiste hinzugekommen, in welche wichtige und häufig genutzte Funktionen ausgelagert wurden.

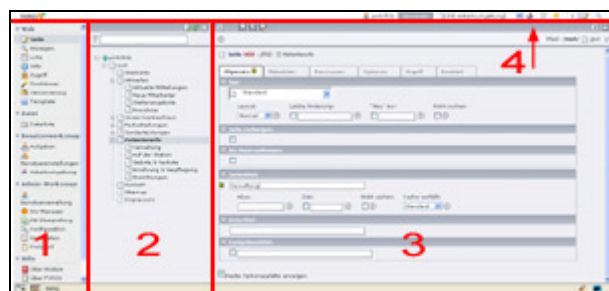


Abb. 1 - Oberfläche des Backend des WCMS TYPO3

Der Aufbau der Oberfläche unterteilt in Modul- (1.) und Navigationsleiste (2.), sowie Detailansicht (3.) und der neuen Kopfleiste (4.) wird in Abbildung Abb. 1 verdeutlicht.

2.3. Situationsanalyse

2.3.1. Auftraggeber / Auftragnehmer

Auftraggeber ist die Klinikum Schönebeck GmbH²⁴, ein Krankenhaus mit über hundertjähriger Geschichte gegründet von Dr. Oskar Pfeil-Schneider im Jahre 1903.

²³ Ein Inlineframe (auch iframe) ist ein HTML-Element, das der Strukturierung von Webseiten dient. Es wird benutzt, um andere Webinhalte als selbständige Dokumente in einem Unterfenster des Browsers anzuzeigen. - o.V. - Inlineframe - <http://de.wikipedia.org/wiki/IFrame> - Stand 17. August 2009

²⁴ o.V. - Klinikum Schönebeck: Startseite - <http://www.kreiskrankenhaus-schoenebeck.de/> - Stand 10. August 2009

Aktuell hat das Krankenhaus über 500 Beschäftigte, davon circa 60 Ärzte und über 250 Schwestern und Pfleger mit überwiegend Staatsexamen.²⁵

Auftragnehmer ist die mediatack GmbH, ein Dienstleister für Internet-Technologien, Online-Lösungen, Marketingservice und Lettershoparbeiten mit Sitz in Chemnitz²⁶.

2.3.2. Aktuelle Online-Präsentation Klinikum Schönebeck GmbH

Der Internetauftritt der Klinikum Schönebeck GmbH besteht seit 2005 und ist aus technischer Sicht sowie aus Gründen des Designs nicht mehr aktuell und bedarf einer Neustrukturierung.

Die Internetpräsenz beruht auf einem veralteten Tabellenlayout und Barrierefreiheit beziehungsweise Barrierearmut wird nicht erreicht. Auch die Trennung von Inhalt und Layout ist nicht gegeben und die Verwaltung und Pflege alter wie neuer Inhalte kann nicht durch die Klinikum Schönebeck GmbH selbst durchgeführt werden.

Das Design hingegen besitzt keine Mängel im Sinne von Fehlern, die nach aktuellem Stand des Wissens unbedingt behoben werden müssen. Im Zuge der Neustrukturierung wird es nur modernisiert und aufgefrischt. Die CI der Klinikum Schönebeck GmbH wird hierbei erhalten und ein Wieder erkennen wird gewährleistet.

Ein bereits bestehendes Intranet existiert nicht und soll daher von Grund auf neu erschaffen werden.

2.4. Pflichtenheft

„Ziel des Projektes soll der Relaunch von Internet und Implementierung des Intranet der Klinikums Schönebeck GmbH sein. Dabei wird beides der zentrale Dreh- und

²⁵ o.V. - Klinikum Schönebeck: Geschichte - <http://www.kreiskrankenhaus-schoenebeck.de/unser-krankenhaus/geschichte.html> - Stand 10. August 2009

²⁶ o.V. - mediatack GmbH - <http://www.mediatack.de> - Stand 18. August 2009

Angelpunkt der internen wie externen Kommunikation, an der mehrere Gruppen in unterschiedlichen Rollen aktiv werden können.“²⁷

Beide Systeme sollen auf Basis von TYPO3 entstehen, um die Pflege beider Kommunikationsmittel einfach zu halten. Hierdurch ergibt sich ein Einsparungspotential aufgrund der wegfallenden Agenturleistung der Inhaltspflege. Alle Inhalte sollen vom Auftraggeber selbst gepflegt werden können, ohne dass dafür Programmierkenntnisse erforderlich sind.

2.4.1. Relevante Punkte

2.4.1.1. TYPO3 Basisentwicklung

HTML-Templates

„Um einen zukunftsfähigen, modularen Aufbau zu gewährleisten, ist auf eine Trennung von Design und Inhalt zu achten.

Dies wird erreicht durch den Einsatz von HTML Designvorlagen, welche im einheitlichen Corporate Design unter der Berücksichtigung der technischen Umgebungsfaktoren XHTML konform erstellt werden.“

Bei der Entwicklung der Designvorlagen wird auf eine XHTML konforme Umsetzung geachtet. Aufgrund der Suchmaschinenoptimierung ist es wichtig, dass diese dem Webstandard des World Wide Web Consortiums entsprechen.

Dies kann durch die Validierung mit dem W3C Markup Validation Service unter <http://validator.w3.org> überprüft werden, sie muss bei einem Test auf Doctype XHTML 1.0 Transitional valide ausfallen. Der Doctype der Seite wird auf XHTML 1.0 Strict eingestellt.

²⁷ Beyer, Thomas - Relaunch Internet + Intranet - Pflichtenheft - 2008

Der strukturelle Aufbau ist sowohl bei der Webpräsenz wie auch bei dem Intranet in ein mehrspaltiges Design gegliedert. Während bei dem Intranet durchgängig drei Spalten verwendet werden, sind es bei der Webpräsenz überwiegend vier, sowie bei beiden Projekten jeweils ein Headerbereich mit eigener Untergliederung.

Browserkompatibilität

Die Seitenausgabe wird für den Internet Explorer Version 6 und 7 sowie Firefox Version 2.0.0.3 auf dem Betriebssystem Windows XP Professional sowie den Safari Version 2.0 auf Macintosh optimiert.

Mehrsprachigkeit

In erster Instanz sind sowohl die Webpräsenz als auch das Intranet in deutscher Sprache zu entwickeln. Die Möglichkeit der Erweiterbarkeit auf andere Sprachen soll gegeben bleiben.

2.4.1.2. TYPO3 Extensions²⁸

DAM (Digital Asset Management)²⁹

Über das Digital Asset Management werden Dokumente und Downloads verwaltet.

Zur umfangreichen Dokumentenverwaltung für eine große Anzahl an Dateien wird die Extension Digital Asset Management eingebunden.

Sie bieten Funktionalitäten wie eine zentrale Speicherung von Dateien sowie die Vergabe von Meta-Informationen. Dies wird durch die Indizierung ermöglicht.

Neben der Indizierung bietet das DAM die Möglichkeit, Inhalte über ordnerähnliche Strukturen zu selektieren und diese Selektionen (Assets) in einem Vorgang zu bearbeiten.

²⁸ Extensions in TYPO3 sind eigenständige Programmteile oder Erweiterungen, welche das WCMS um weitere Funktionen ergänzen

²⁹ o.V. - Media (DAM) - <http://typo3.org/extensions/repository/view/dam/current/> - Stand 08. August 2009

Selektionen lassen sich zudem durch die Angabe von Suchkriterien weiter verfeinern. So behält der TYPO3-Anwender stets den Überblick über seinen Dateibestand und kann selbst bei einer Vielzahl an Dateien noch hocheffizient arbeiten.

Backend Administratoren haben Zugriff auf einen zentralen Ordner, in welchem interne Dokumente und Downloads verwaltet werden.

Indexed Search (Volltextsuche)

Mit Hilfe dieser Extension wird eine indizierende Suche implementiert. Indizierend bedeutet in diesem Fall, dass die Seiten der Internetpräsenz in TYPO3 eingelesen und in der Datenbank vorgehalten werden, um ein schnelles und ressourcensparendes Ergebnis zu gewährleisten.

Über eine zentrale Suche können die Inhalte im Hauptinhaltsbereich der Seite gefunden werden.

cal (Terminkalender)³⁰

Im Kontextbereich jeder Seite wird eine Übersicht des aktuellen Monats eingebunden. Tage mit Veranstaltungen werden auf dieser farblich hervorgehoben.

Über Schaltflächen ist es möglich, jeweils zum nächsten beziehungsweise letzten Monat zu navigieren.

Über die Monatsansicht kann in die Tage hinein navigiert werden. Der aktuelle Tag wird hierbei optisch hervorgehoben.

Es werden im Veranstaltungskalender nur diejenigen Termine angezeigt, die der aktuellen Rubrik in der Navigation entsprechen.

³⁰ o.V. - Calendar Base - <http://typo3.org/extensions/repository/view/cal/current/> - Stand 06. August 2009

2.4.1.3. Seiten- und Rechtestruktur des Intranet

Das Intranet soll folgende Bereiche erhalten:

- Allgemeiner Bereich
- Fachabteilungen

Der allgemeine Bereich ist für jeden Nutzer, unabhängig seines Fachbereichs, zugänglich und frei einsehbar. News und Termine aus den einzelnen Fachabteilungen werden hier gesammelt noch einmal gelistet.

Des Weiteren erhält jede Fachabteilung einen eigenen Bereich. Der Zugang zu diesem soll eingeschränkt für jedermann möglich sein. Mitglieder der Fachabteilung erhalten uneingeschränkten Zugang. Für jede Fachabteilung gibt es Redakteure, welche innerhalb Ihres Bereiches Inhalte bereitstellen können.

3. Umsetzung

3.1. System aufsetzen

Ursprünglich war eine Verwaltung beider Systeme über eine TYPO3 Installation vorgesehen. Es sollte möglich sein, dass ein Redakteur oder Administrator über einen Zugang gleichzeitig die Webpräsenz wie auch das Intranet betreut oder Inhalte einpflegt.

Um dies zu ermöglichen, wurde ein gemeinsames Web mit installiertem TYPO3 für beide Projekte angelegt. Innerhalb von TYPO3 wären diese über zwei parallele Menübäume und dem Einsatz von Multidomains³¹ im Frontend getrennt. Dies wurde jedoch im Laufe der Projektentwicklung geändert. Beide Projekte laufen nach wie vor mittels desselben WCMS, jedoch jedes Projekt auf einem eigenen Web mit eigener TYPO3-Installation.

Somit wird die Internetpräsenz und das Intranet in ihrer Entwicklungsphase auf einem Server des Anbieters Mittwald CM Service GmbH & Co KG³², nachfolgend Mittwald genannt, gehostet. Nach Abschluss der Arbeit und Abnahme des Intranets durch den Kunden soll das Intranet auf einen unternehmenseigenen Server portiert werden. Die Internetpräsenz hingegen wird auch in Zukunft auf dem Mittwald-Server bleiben und nicht umziehen.

Das Aufsetzen der Serverumgebung selbst ist durch Nutzung der Dienste des Anbieters Mittwald einfach zu bewerkstelligen. Das Unternehmen mediatack GmbH besitzt hierfür einen „Reseller“-Zugang und kann eigene Webs anlegen. TYPO3 wird bei der Einrichtung bereits mit installiert. Die technischen Anforderungen des WCMS werden hierbei alle erfüllt.

³¹ Innerhalb von TYPO3 ist es möglich, einzelnen Seiten oder Seitenbäumen unterschiedliche URL's zuzuweisen. Dadurch können mehrere im Frontend getrennte Seiten über eine einzelne TYPO3 Installation verwaltet werden.

³² o.V. - Mittwald - <http://www.mittwald.de> - Stand 05. August 2009

3.2. Einrichten des Systems

Bereits kurze Zeit nach Einrichtung ist das System betriebsbereit und der Backendzugang mit Administratorrechten³³, Zugang zur Datenbank mittels phpMyAdmin³⁴ und ein FTP-Zugang stehen zur Verfügung, um mit der technischen Umsetzung beginnen zu können.

Die in den folgenden Abschnitten beschriebenen Einstellungen, Erweiterungen und Konfigurationen sind die Grundlage für das weitere Arbeiten mit dem WCMS und wurden für beide Systeme (Intranet und Webpräsenz) getrennt durchgeführt.

Die Webpräsenz war das erste eingerichtete Web, da in der ursprünglichen Planung beide Systeme hierüber laufen sollten. Für das später eingerichtete Web des Intranets konnten die gewählten Einstellungen der Webpräsenz jedoch in weiten Teilen übernommen werden. In der Grundkonfiguration ähneln sich somit beide Systeme.

3.2.1. Benutzereinstellungen

Der erste Schritt nach dem Einloggen in das System ist normalerweise die Einstellung der Sprache für den Benutzerzugang. Standardmäßig ist das Backend von TYPO3 auf englische Sprache eingestellt. Über den Punkt Benutzerverwaltung kann dieser Wert verändert werden. Bei der Mittwald-Installation ist deutsch jedoch bereits vorausgewählt, so dass diese Einstellung unangetastet gelassen werden kann.

3.2.2. Extensions

Der nächste wichtige Punkt ist die Erweiterung des Systems durch so genannte Extensions. Diese werden meist von anderen Entwicklern des WCMS bereitgestellt

³³ dieser Zugang besitzt uneingeschränkte Rechte auf alle Funktionen

³⁴ eine Weboberfläche zur Verwaltung der MySQL Datenbank

und ergänzen das System um viele nützliche Funktionen wie zum Beispiel um einen Kalender oder News.

Die Installation erfolgt über den Extension Manager. Über diesen kann man einerseits neue Erweiterungen herunterladen und installieren sowie bereits installierte Erweiterungen auf Updates überprüfen und diese auch durchführen. Der Manager verfügt ebenfalls über ein Dependency-Checker. Dies bedeutet, dass bei Installation oder Upgrade überprüft wird, ob zum reibungslosen Betrieb weitere Extensions benötigt werden oder es zu Konflikten mit bereits installierten Erweiterungen kommt. In beiden Fällen kann die Installation oder Deinstallation entsprechender Extensions in die Wege geleitet werden.

Die durch das Pflichtenheft vorgegebenen Extensions wie das Dateienverwaltungssystem „DAM“, der Kalender „cal“ oder die Suche „Indexed Search“ wurden jetzt installiert. Genauer wird auf ihre Konfiguration und Anpassung für das System zu einem späteren Zeitpunkt eingegangen (Kapitel 3.5).

Eine Extension zum Thema „Template Building“ wurde hier ebenfalls installiert. In TYPO3 konkurrieren 2 Erweiterungen, um das Arbeiten mit Templates zu unterstützen. Erstens der „Page Template Selector“ (rlmp_tmplselector)³⁵ für die Vorgehensweise des „Modern Template Building“ und zweitens „TemplaVoila!“ (templavoila)³⁶ für das „Future Template Building“.

Variantenanalyse

Beide Extensions wurden von Kasper Skårhøj und Robert Lemke erstellt. Der „Page Template Selector“ wurde ursprünglich entwickelt, um dem Redakteur oder Backenduser die Möglichkeit zu geben, jeder Seite individuell ein HTML-Template zu zuweisen. Der Programmierer stellt dem Redakteur eine Auswahl an HTML-Designvorlagen zur Verfügung, aus der er wählen kann. Dieser muss nun

³⁵ o.V. - Page Template Selector - http://typo3.org/extensions/repository/view/rlmp_tmplselector/current/ - Stand 18. August 2009

³⁶ o.V. - TemplaVoila! - <http://typo3.org/extensions/repository/view/templavoila/current/> - Stand 18. August 2009

selbst keine Anpassungen an den HTML-Templates vornehmen - eine Aufgabentätigkeit, für die ein Redakteur im Regelfall auch keine Berechtigung erhält.

Die Entwicklung von „TemplaVoila!“ setzte sich mit der Problematik auseinander, wie man komplexere Seitenstrukturen erzeugen kann, als es mit dem typischen Spaltendesign, welches TYPO3 standardmäßig zugrunde liegt, möglich ist.

Die Entscheidung in dieser Arbeit fiel auf die Verwendung des „Page Template Selectors“. Die Vorteile von „TemplaVoila!“ bezüglich der komplexeren Seitenstrukturen spielen in dieser Arbeit keine Rolle, da das Design der beiden zu entwickelnden Seiten die Spaltenaufteilung von TYPO3 unterstützt. Die geringere Komplexität und einfachere Anwendung vom „Page Template Selector“ gaben schlussendlich die Entscheidung.

Nach Installation aller zu verwendeten Erweiterungen wurden diese über den Extensionmanager³⁷ anschließend auf Verfügbarkeit der deutschen Sprache überprüft und diese aktualisiert.

3.3. TYPO3- und HTML-Templates

Das Design der Seite wird mittels HTML und CSS gesteuert. Obwohl es grundlegend möglich ist, die Struktur der Seite komplett über das TYPO3-Template mittels TypoScript zu steuern, empfiehlt sich der Einsatz von HTML-Templates, da dies die Arbeit bei größeren Projekten sehr stark vereinfacht.

Des Weiteren wird die Nutzung von HTML-Templates für den Seitenaufbau durch die im vorherigen Kapitel installierte Extension „Page Template Selector“ vorgegeben. Durch Verwendung dieser Extension ist es möglich, schnell und einfach weitere Designvorlagen dem Redakteur zur Auswahl anzubieten.

Ebenso wird die Aufteilung des HTML-Templates in Main und Sub ermöglicht.

³⁷ Modul im Backend von TYPO3 zum Verwalten aller Erweiterungen.

In dieser Arbeit wurde an dieser Stelle mit der Umsetzung des Designs in HTML-Templates begonnen. Anschließend wurden die TYPO3-Templates definiert, welche die Inhalte des WCMS mit den HTML-Templates verbinden.

3.3.1. HTML-Templates

Die HTML Designvorlagen enthalten den Grundaufbau der Seite. Die Struktur- und Designvorgaben des Pflichtenheftes müssen hierbei erfüllt werden. Nach aktuellem Stand wird die Struktur der Seite mittels DIV's³⁸ generiert. Über die Verwendung von ID's werden diese anschließend durch CSS formatiert.

Inhalt enthalten diese HTML-Templates keinen. Stattdessen sind Marker enthalten, welche dem TYPO3 Template vorgeben, wo welcher Inhalt anschließend platziert werden soll.

Bei der Entwicklung der HTML-Designvorlagen wurden diese jedoch in der ersten Version mit Inhalt gefüllt. Mittels dieser „Test“-Inhalte war es möglich, die CSS-Formatierung der Seite in einem Maße anzupassen, wie es ohne diese Inhalte nicht möglich gewesen wäre.

3.3.1.1. Webpräsenz

Das HTML-Template für die Webpräsenz orientiert sich in seiner Entwicklung an den Vorgaben durch das Design. Das Design liegt im Format PSD³⁹ vor und gibt die Struktur der Seite indirekt vor. Die Entscheidung, welche Struktur sich aus den Vorgaben ergibt, liegt beim Entwickler selbst und



Abb. 2 - Grundstruktur der Webpräsenz

³⁸ ein HTML Element/Tag, der einen Block/Bereich kennzeichnet

³⁹ Dateiformat für Grafikdateien von Adobe Photoshop

sollte ein leichtes Bearbeiten der Inhalte zum Ziel haben. Anhand der roten Linien im Bild (Abb. 2) kann man erkennen, wie in diesem Fall die Struktur mittels DIV's umgesetzt wurde.

Grob unterteilt wird die Seite in:

- eine Servicenavigation (1),
- ein Flashheader⁴⁰ (2),
- die Hauptnavigation (3),
- die Subnavigation (4),
- ein navigationsbezogenes Grafikelement (5),
- der Contentbereich (6),
- ein weiterer Inhaltsbereich für zusätzliche Informationen (7)
- und abschließend die Footernavigation (8).

Die Einteilung der HTML-Designvorlagen in Main und Sub wurde hier genutzt, um verschiedene Sub-„HTML-Templates“ zu erzeugen, welche die Seite zum Teil beeinflussen. Die Elemente 1, 2, 3 und 8 bilden das Rückgrat der Webpräsenz und sind auf allen Seiten identisch. Aus diesem Grund werden diese über das Main-„HTML-Template“ generiert (Anhang 1). Im mittleren Bereich, in dem sich die Elemente 4, 5, 6 und 7 befinden, wird an dieser Stelle das Sub-„HTML-Template“ (Anhang 2) integriert. In Abb. 2 ist die Standardvariante zu sehen, welche überwiegend in der Internetpräsenz zum Einsatz kommt. Es gibt jedoch noch zwei weitere Alternativen, die den Redakteuren ebenfalls zu Verfügung stehen.

Zuerst wird auf der Startseite ein Sub-„HTML-Template“ verwendet, in welchem die Inhaltsbereiche 4, 5 und 6 zu einem zusammengefügt sind (Anhang 3). Eine Subnavigation sowie die Grafik fehlen hier und der Contentbereich erstreckt sich fast über komplette Breite. Ein weiteres Sub-„HTML-Template“ verbindet die Inhaltsbereiche 6 und 7 und vergrößert so den Contentbereich nach rechts (Anhang 4).

⁴⁰ Ein Kopfelement einer HTML Seite, welches einen Flashfilm enthält

Innerhalb all dieser Bereiche kommen Marker zum Einsatz, über welche mittels des TYPO3 Templates die Inhalte anschließend positioniert werden.

3.3.1.2. Intranet

Wie auch bei der Vorbereitung des Webs profitiert die HTML Templateerstellung des Intranets von zuvor fertig gestellten HTML-Templates der Webpräsenz. Auch wenn sie nicht identisch sind, so ist doch eine ähnliche



Abb. 3 - Grundstruktur des Intranets

Grundstruktur vorhanden, was ein Anpassen der bereits existierenden Arbeit ermöglicht. Die Struktur basiert beim Intranet auf nur 3 Spalten, und diese Zahl ändert sich auch nicht durch den Einsatz verschiedener Sub-„HTML-Templates“. Im Groben besteht das Intranet aus (Abb. 3):

- dem Header (1),
- der Navigation (2),
- dem Contentbereich (3),
- einer weiteren Spalte für Zusatzinformationen (4)
- sowie aus einem Footer (5).

Die Inhalte werden ebenfalls wieder über Marker in den einzelnen Bereichen positioniert.

3.3.2. TYPO3 Template

Das TYPO3 Template ist essentieller Bestandteil einer Webseite und zwingend notwendig. Auch wenn Inhalte bereits über das Backend angelegt werden könnten, ohne Verwendung des Templates gibt es keine Vereinbarungen, wie und in welcher Form die Inhalte auf der Webseite präsentiert werden sollten.

Das Template ist ein Datensatz und muss einer Seite zugewiesen werden. Da bei einem frisch installierten TYPO3 noch keine Seite existiert, muss diese zu erst angelegt werden. Es wurde die Seite „kkh“ erstellt und dieser „Wurzel“-

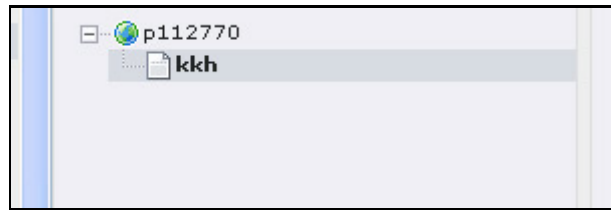


Abb. 4 - die erste Seite in einem neuen TYPO3 Projekt

Seite⁴¹ wurde der neue Datensatz „Template“ anschließend zugewiesen (Abb. 5). Alle Kindelemente dieser Seite erben anschließend auch das Template und unterliegen somit denselben Vereinbarungen. Jedoch können auf Unterseiten wiederum neue Templates angelegt werden, welche das Verhalten der Seiten neu beeinflussen.

Das Template muss nun noch mittels der TYPO3 eigenen Scriptsprache TypoScript angepasst werden. Hierfür enthält

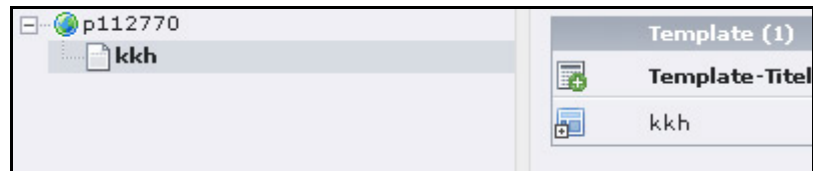


Abb. 5 - eine „Wurzel“-Seite mit dem TYPO3-Template

der Datensatz das Feld „Konfiguration“, in welches die Anweisungen geschrieben werden können. Das Bearbeiten der TypoScript-Anweisungen wird jedoch vereinfacht, wenn man sie auslagert.

Mittels folgendem Code ist es möglich, TypoScript aus externen Dateien zu importieren.

```
<INCLUDE_TYPOSCRIPT: source="FILE:
fileadmin/kkh/template/ts/setup_page.ts">
```

Der TypoScript Code wird in eine Textdatei ausgegliedert und kann dann mittels eines Texteditors bearbeitet werden. Einige Texteditoren bieten mittlerweile auch Syntax-Hervorhebung für TypoScript an, was das Arbeiten durchaus vereinfachen kann.

⁴¹ Diese Seite ist Einstiegseite in den Seitenbaum. Sie ist in dieser Arbeit die höchste Ebene unterhalb der Weltkugel.

Innerhalb der TypoScript-Anweisungen ist die Definition des PAGE-Objektes der wichtigste Punkt. Dieses stellt den Ausgangspunkt des hierarchischen Informationsbaums dar und wird daher auch als Toplevel Objekt (TLO) bezeichnet⁴².

Durch folgenden Code wird nun ein PAGE-Objekt mit dem Bezeichner „page“ erzeugt:

```
page = PAGE
```

Des Weiteren besitzt das PAGE-Objekt einige Eigenschaften, welche die Headerdaten des HTML Dokumentes definieren. Diese werden hier exemplarisch aufgezeigt.

```
meta {
    keywords.ifEmpty (
        Klinikum Schönebeck
    )
}

includeCSS {
    file1 = fileadmin/kkh/template/css/kkh.css
}

headerData {
    10 = TEXT
    10.value (
        <meta name="robots" content="index, follow"
/>
        <meta name="language" content="de" />
    )
}
```

Anschließend erfolgt der wichtigste Punkt: die Zuweisung von Inhalten an die Marker. Die Umsetzung erfolgt mittels folgendem Code:

⁴² vgl. Andreas Stöckl, Frank Bongers (2006): Einstieg in TYPO3 4.0; 2., aktualisierte und erweiterte Auflage 2006, Seite 379

```
10 = TEMPLATE
10 {
    template < plugin.tx_rlmptmplselector_pi1
    template.templateType = main
    workOnSubpart = DOCUMENT
    marks {
        SUBTEMPLATE < subtemplate
        HEADERLINK < headerlink
        SEARCHBOX < searchbox
        SERVICENAVIGATION < servicemenu
        FLASH < flash
        NAVIGATION < mainmenu
        FOOTER < footer
    }
}
```

Um auf die Funktionalität des Page Template Selectors (Kapitel 3.2.2) zurückzugreifen, wird dieser in das Objekt „template“ kopiert, welchem nun alle relevanten Informationen zur Verfügung stehen - unter anderem der Speicherort der Main und Sub HTML-Designvorlagen.

Des Weiteren wird dem Objekt „template“ mitgeteilt, dass sich die folgenden Angaben auf das Main HTML-Template beziehen und dass der Bereich, welcher für die Extension interessant ist, mit dem Marker DOCUMENT gekennzeichnet ist.

Alle im HTML-Template verwendeten Marker werden nun nachfolgend aufgelistet und es wird ihnen ein TypoScript Objekt zugewiesen. Diese Objekte werden ebenfalls in TypoScript definiert und sind letztendlich dafür verantwortlich, welche Inhalte anstelle der Marker eingebunden werden sollen.

Zur Verdeutlichung ist im Anhang (Anhang 5) der TypoScript Code ausführlicher hinterlegt.

3.4. Grundstruktur und Seitenaufbau

Nachdem die Templates erstellt wurden, ist der nächste Schritt das Anlegen der Seitenstruktur. Hierfür werden der zuvor erstellten „Wurzel“-Seite neue Seiten untergeordnet, welche einerseits das Template erben und andererseits den Seitenbaum des Webauftritts darstellen. Die „Wurzel“-Seite selbst ist kein Teil der Webseite an sich, sie dient nur als Einstiegspunkt. Um zu verhindern, dass diese Seite im Frontend auftaucht, wird der Typ der Seite auf „Verweis“ geändert und auf eine Unterseite verlinkt.

3.4.1. Anlegen der Ordner- und Seitenhierarchie

Das Anlegen der benötigten Seiten innerhalb TYPO3's geht sehr einfach von statten. Mithilfe eines Assistenten ist es schnell und einfach möglich, neue Seiten zu erstellen und innerhalb der Baumstruktur zu positionieren. TYPO3 stellt auch im Modul Funktionen eine Möglichkeit zur Verfügung, um gleichzeitig mehrere Seiten zu generieren.

Auch ist es möglich, Seiten von unterschiedlichem Typ herzustellen. Den Typ des Verweises und seine Funktionalität wurde bereits im vorherigen Kapitel kurz erläutert. Ein weiterer wichtiger Typ zur Gestaltung der Baumstruktur und der angestrebten Funktionalität ist der Systemordner. Im Prinzip ist es ein Datensatz wie die normale Seite auch, jedoch wird er im Frontend nicht dargestellt.

Genutzt wird er in dieser Arbeit zu zwei unterschiedlichen Zwecken. Einerseits innerhalb des Seitenbaums um bestimmte Unterseiten aus der Hauptnavigationsstruktur auszuschließen und gesondert zu behandeln. Dies sind zum Beispiel Seiten wie Impressum, Kontakt oder Suche (Abb. 6. - Servicemenu).

Andererseits um parallel zum obersten Seitenelement, welches den Einstiegspunkt der Webseite im Frontend repräsentiert, eine Ordnerstruktur zu errichten, in der Datensätze verwaltet werden (Abb. 6. - Datenmanagement). Die hier zu verwaltenden Datensätze sind in der Regel Inhalte und Informationen, die



Abb. 6 - erste Ebene der Baumstruktur der Webpräsenz..

nicht zwangsweise nur einer bestimmten Seite im Baum zugeordnet werden müssen. Während Inhalte wie Texte direkt auf den jeweiligen Seiten verwaltet werden, so werden Inhalte wie zum Beispiel News oder Adressen nicht nur auf einer Seite veröffentlicht, sondern können mehrfach eingebunden werden. Häufig sind dies auch Datensätze, die nicht direkt auf der Seite veröffentlicht werden können, sondern über ein Inhaltselement - welches veröffentlicht wird - auf den Seiten eingebunden werden.

Dies ist keine von TYPO3 vorgeschriebene Herangehensweise, aber sie ermöglicht es, die Datensätze besser zu strukturieren. Speziell im Fall des Intranets ist es so auch möglich, Zugangs- und Bearbeitungsrechte zum Beispiel für Datensätze wie News den Redakteuren zu gewährleisten, ihnen aber keinen Zugang zu der Seite zu geben, auf welcher das hierfür benötigte News-Plug-in⁴³ installiert ist. So ist es ihnen möglich, neue News anzulegen, sie haben aber keinen Einfluss darauf, wie und wo diese News im Frontend ausgegeben werden.

3.4.1.1. Webpräsenz

Die konkrete Baumstruktur der Seite ergibt sich aus den benötigten Seiten selbst - vorgegeben durch die alte Webpräsenz der Klinikum Schönebeck GmbH - und der gewünschten Navigationsstruktur. Die Hauptnavigation enthält hierbei alle wichtigen Seiten. Dazu gehören (Abb. 6):

- Aktuelles
- Unser Krankenhaus

⁴³ Plug-in - ist ein von einer Extension bereitgestelltes Inhaltselement

- Zentren
- Kliniken
- Funktionsbereiche
- Medizinische Versorgungszentren
- Sonderleistungen
- Patienteninfo

Mit dieser Hauptnavigation und allen untergeordneten Ebenen bilden sie das Grundgerüst der Webpräsenz.

Ausgliedert aus der Hauptnavigation mittels eines Systemordners werden hingegen folgende Seiten (Abb. 7):

- Sitemap
- Impressum
- Anfahrt
- Kontakt
- Startseite
- Seite weiterempfehlen
- Suche

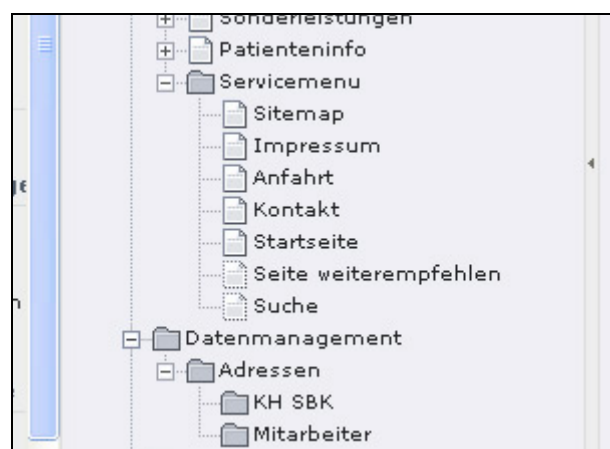


Abb. 7 - Auszug aus Seitenbaum - Service und Datenmanagementordner aufgeklappt

Diese werden überwiegend durch die im Frontend vorhandene Servicenavigation angesprochen.

In der Parallel zum Seitenbaum angelegten Ordnerstruktur werden Adressdatensätze gesammelt (Abb. 7 - Datenmanagement).

3.4.1.2. Intranet

Die Seitenstruktur des Intranets ist im Gegensatz zur Webpräsenz relativ einfach gehalten (Abb. 8). Alle im Frontend anzuzeigenden Seiten sind direkte Unter- oder Unter-Unterseiten des obersten Seitenelementes. Es kommen keine Systemordner zum ausgliedern einiger Seiten aus der Hauptnavigation vor. Eine Servicenavigation kommt nicht zum Einsatz.

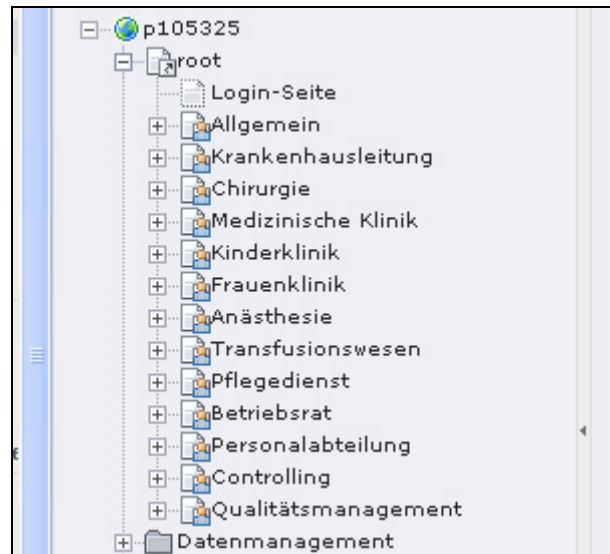


Abb. 8 - Seitenbaum des Intranets - geschützt

Sehr viel umfangreicher als bei der Webpräsenz ist hingegen die Ordnerstruktur zur Verwaltung von Daten wie News oder Terminen. Ihr Aufbau richtet sich nach der vorgegebenen Seitenstruktur und der Vorgabe bezüglich der Rechtevergabe der unterschiedlichen Redakteursgruppen.

Zu diesem Zeitpunkt wurde jedoch erst eine sehr rudimentäre Ordnerstruktur geschaffen. Die Ordnerstruktur innerhalb des Intranets hat im Laufe der Entwicklung viele Änderungen erfahren und wurde mehrfach neu strukturiert. Ursache hierfür ist der enge Zusammenhang mit der komplexen Rechteverwaltung des Intranets.

3.5. Module (Frontend)

Einer der größten bereits erwähnten Vorteile des WCMS TYPO3 ist die modulare Erweiterbarkeit des Funktionsumfanges mittels Extensions. Diese ermöglichen ein breites Spektrum an neuen umfangreichen Funktionalitäten, um den Mehrwert der Webseite oder des Intranets zu steigern.

Ein einfaches Herunterladen und Installieren führt jedoch bei den wenigsten Erweiterungen zu einem sauberen Ergebnis. Überwiegend müssen sie erst umfangreich eingebunden und angepasst werden.

In den folgenden Kapiteln wird auf die wichtigsten verwendeten Erweiterungen eingegangen und mögliche Alternativen werden aufgezeigt. Die Installation der meisten Extensions ist zu diesem Zeitpunkt bereits durchgeführt (Kapitel 3.2.2), jedoch wurden bisher keine Anpassungen vorgenommen.

3.5.1. Terminkalender

Der Terminkalender kommt derzeit nur im Intranet zum Einsatz.

3.5.1.1. Anforderungen

Die Mindestanforderungen an eine Extension zum Einsatz als Kalender im Intranet sind in erster Linie folgende:

- Pflege der Termine über das Backend
- eine Listenansicht zukünftiger Termine
- eine kleine Monatsansicht zur Vorschau im rechten Contentbereich (Abb. 3 - Punkt 4)
- eine Detailansicht eines gewählten Termins
- Kategorisierung von Terminen - Dieser Punkt ist essentiell wichtig, um die gewünschte Rechtestruktur des Intranets gewährleisten zu können.

Ein weiterer wichtiger Punkt ist der Support. Es gibt zwar keine offiziellen Anlaufstellen - die Verbreitung und Nutzung der Extension in der Community gibt aber Aufschluss über die Menge an Informationsmaterial in Form von Tutorials oder hilfreichen Forumsbeiträgen, die bei der Installation und Konfiguration weiterhelfen.

3.5.1.2. Variantenanalyse

Die Wahl der Extension für den Kalender ist bereits durch das Pflichtenheft vorgegeben. Nachfolgend sind Alternativen und Gründe aufgelistet, weshalb

letztendlich die Entscheidung für die in dieser Arbeit verwendete Extension gefallen ist.

JW Calendar (jw_calendar)⁴⁴

Diese Extension wurde von Jens Witt verfasst. Sie liegt aktuell in der Version 1.3.20 vor und hat den Status „beta“.

Vorteil:

Die von mir gewünschten Mindestanforderungen werden erfüllt. Ihrer Anleitung⁴⁵ sind folgende Funktionen zu entnehmen:

- Ein Kalender mit Listen und Monatsansicht, einer Tagesansicht und Einzelansicht der Ereignisse.
- Ereignisseinträge im Frontend möglich
- Kategorienansicht und FE⁴⁶ Ereignisseinträge mit Zugriffskontrolle
- FE Einträge können direkt ausgegeben oder vorher geprüft werden
- Benachrichtigung per Mail wenn FE Eintrag erfolgte
- Unterstützung von sich wiederholenden Ereignissen
- Ausnahmeereignisse um bestimmte Bereiche in der Monatsansicht hervorzuheben und/oder die Anzeige von sich wiederholenden Ereignissen zu steuern
- Kombinierte Anzeige von z.B. Monatsansicht und einer Tagesansicht des Tages der in der Monatsansicht selektiert wird auf einer Seite

Nachteil:

Der schwerwiegendste Nachteil dieser Extension ist Ihre Aktualität. Zuletzt aktualisiert wurde sie am 01.03.2005. Durch diesen Umstand kann eine Kompatibilität mit der aktuellen Version des WCMS TYPO3 nicht zu 100% sichergestellt werden. Auch Support in Form von Newsbeiträgen oder Foren ist selten aktuell.

⁴⁴ o.V. - JW Calendar - http://typo3.org/extensions/repository/view/jw_calendar/current/ - Stand 08. August 2009

⁴⁵ o.V. - JW Calendar - http://typo3.org/documentation/document-library/extension-manuals/jw_calendar/1.3.20/view/1/1/#id3554775 - Stand 08. August 2009

⁴⁶ FE - Frontend

Calendar Base (cal)⁴⁷

Die Autoren der „cal“-Extension sind Mario Matzulla, Jeff Segars und Franz Koch. Die Erweiterung liegt aktuell in der Version 1.2.0 vor und hat den Status „stabil“.

Vorteil:

Alle Mindestanforderungen werden durch diese Extension ebenfalls erfüllt. Sie hat ein umfangreiches Repertoire an Ansichtsformen, unter anderem die geforderten Listen-, Monats- und Detailansichten. Kategorien werden ebenfalls unterstützt.

Sie wird ständig weiterentwickelt und wird mittlerweile am häufigsten genutzt. Support in der Community ist hierfür sehr zahlreich vorhanden.

Weiterhin ist diese Extension eine der umfangreichsten Extensions, die es derzeit gibt. Ihre Funktionsvielfalt ist mit keiner anderen Kalender Extension zu vergleichen. Man kann sagen, dass nahezu jede Problemstellung im Zusammenhang mit Kalendern mit dieser Extension bewerkstelligt werden kann.

Nachteil:

Dieser Umfang ist jedoch auch ihr größter Nachteil. Für ein Projekt wie dieses Intranet wird nur 10% der möglichen Funktionsvielfalt effektiv genutzt. Die Komplexität und die damit einhergehende Einarbeitungszeit muss dennoch bewältigt werden.

3.5.1.3. Entscheidung

Schlussendlich fiel in dieser Arbeit die Entscheidung auf die Calendar Base Extension. Ihre Aktualität ist ihr größter Pluspunkt. Sie ist im Moment die umfangreichste Erweiterung auf diesem Gebiet und wird regelmäßig überarbeitet und erweitert. Hinter ihr stehen mehrere Entwickler, die sie voran bringen.

⁴⁷ o.V. - Calendar Base - <http://typo3.org/extensions/repository/view/cal/current/> - Stand 06. August 2009

Im Endeffekt führte dies auch dazu, dass die Alternativen, welche es früher noch zahlreicher gab, in ihrer Weiterentwicklung stoppten. Die daraus resultierenden Nachteile der Alternativen machten die Entscheidung für die Calendar Base Extension endgültig.

Der Funktionsumfang, welcher die Extension so groß und komplex macht, kann später auch zu Vorteilen führen, sollte für das Intranet diesbezüglich neue Anforderungen gestellt werden.

3.5.1.4. Konfiguration

Die Einstellungen zum reibungslosen Betrieb des Kalenders sind umfangreich. Es müssen Seiten für die Listen- wie auch für die Singleansicht angelegt werden, in welche das Kalender-Plug-in eingebunden wird (Abb. 9). Die Ordnerstruktur muss erweitert werden, um die Termine und kalenderrelevante Daten zu speichern (Abb. 10) und das TYPO3 Template muss werden, um die Ausgabe der Ansichten zu bearbeiten und formatieren.

Mittels TypoScript kann überwiegend die komplette Funktionalität der Extension gesteuert werden. Hierbei wird das TypoScript wie bereits in

Kapitel 3.3.2 beschrieben ausgegliedert, um eine Bearbeitung durch einen Editor zu ermöglichen und Änderungen schnell und einfach durchzuführen.

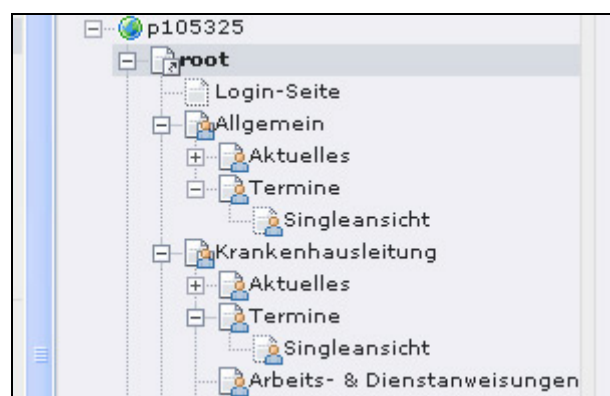


Abb. 9 - Auszug aus dem Seitenbaum des Intranet, mit den Seiten für Termine

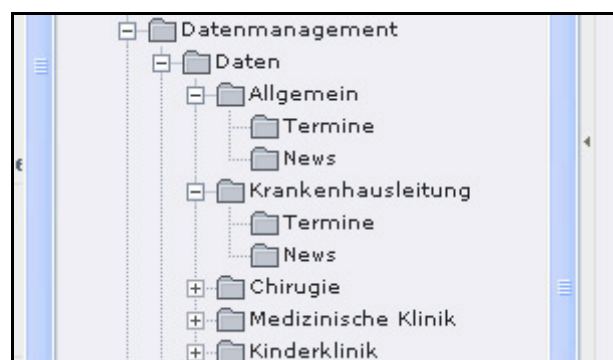


Abb. 10 - Auszug aus der Sysordnerstruktur des Intranet, mit den Ordnern für Termine

Wichtige Werte sind unter anderem:

- *plugin.tx_cal_controller.pidList* - der „Startpunkt“ im Seitenbaum ab welchem die Plug-ins nach erstellten Terminen oder anderen kalenderbezogenen Datensätzen suchen (in dieser Arbeit ein in der Ordnerstruktur angelegter Systemordner, welchem alle kalenderbezogenen Daten untergeordnet sind)
- *plugin.tx_cal_controller.recursive* - bezieht sich auf den „Startpunkt“ und gibt an, wie tief in der Ordnerstruktur gesucht werden soll
- *plugin.tx_cal_controller.view.list.listTemplate* - dieser Wert gibt an, welche HTML-Designvorlage für die Listendarstellung verwendet werden soll
- *plugin.tx_cal_controller.view.list.starttime* - der Startzeitpunkt, welcher in der Listenansicht standardmäßig gewählt sein soll. Im Falle dieser Arbeit ist es „cal:monthstart“ - der erste Tag des aktuellen Monats
- *plugin.tx_cal_controller.view.list.endtime* - der Endzeitpunkt, welcher in der Listenansicht standardmäßig gewählt sein soll. Im Falle dieser Arbeit ist es „cal:monthend“ - der letzte Tag des aktuellen Monats.
- *plugin.tx_cal_controller.view.list.pageBrowser.usePageBrowser* - mittels dieses Werts ist es möglich, die Anzahl gleichzeitig angezeigter Events in der Listenansicht zu reduzieren und alle weiteren durch einen „pagebrowser“ zu erreichen. In dieser Arbeit wird dieser Wert auf 0 gesetzt, da er hier keine Verwendung findet.

Ein wichtiger Punkt beim Kalender im Intranet soll sein, dass nicht auf jeder Seite, auf der das Kalender-Plug-in integriert ist, immer alle News angezeigt werden sollen, sondern nur die für die aktuelle Seite Gültigen. Um dies zu bewerkstelligen, gibt es theoretisch verschiedene Herangehensweisen.

Eine Möglichkeit wäre, die Termine in unterschiedlichen Ordnern zu speichern, um je nach dem wo man sich befindet, auf die unterschiedlichen Ordner zugreifen zu können. Der Nachteil hierbei ist, dass wenn ein Termin doch in zwei unterschiedlichen Bereichen veröffentlicht werden soll, man ihn kopieren und somit zweimal im Backend speichern muss.

In dieser Arbeit wird deshalb auf die Verwendung von Kategorien zurückgegriffen. Die Extension Calendar Base bietet die Möglichkeit, Kategorien für den Kalender anzulegen. Diese müssen ebenfalls in der Ordnerstruktur in oder unterhalb des Ordners definiert werden, den wir bereits als „Startpunkt“ für alle Kalender bezogenen Datensätze definiert haben (Abb. 10). Anschließend ist es möglich, neuen Terminen eine oder mehrere Kategorien zuzuweisen.

Weiterhin muss jetzt noch definiert werden, auf welchem Teil der Seite welche Kategorien angezeigt werden sollen. Dies geschieht auch wieder mittels TypoScript im TYPO3 Template.

```
[PIDinRootline = 60]
# Chirurgie
plugin.tx_cal_controller.view.categoryMode = 1
plugin.tx_cal_controller.view.category = 3
[PIDinRootline = 61]
# Qualitätsmanagement
plugin.tx_cal_controller.view.categoryMode = 1
plugin.tx_cal_controller.view.category = 13
[end]
```

Die Definition innerhalb der eckigen Klammern ist hierbei eine TypoScript „if else“-Anweisung und besagt, dass die Wertzuweisungen innerhalb nur dann ausgeführt werden, wie sich die aktuelle Seite im Frontent gerade auf PageID 60 befindet, beziehungsweise ein Kindelement davon ist. Ist dies der Fall, wird der categoryMode auf „1“ gesetzt („1“ bedeutet in diesem Fall, dass die Auswahl der Kategorie bindend ist. „2“ würde eine Invertierung bedeuten - zeige alle Kategorien außer der gewählten, und „0“ würde bedeuten, dass die hier nachfolgende Kategorieauswahl ohne Bedeutung ist) und die anzuzeigende Kategorie auf „3“. Sollte dies nicht der Fall sein, findet die Überprüfung auf die PageID 61 statt.

Diese „if else“-Anweisung muss für jeden Teilbereich der Seite fortgesetzt werden, in welchem eine eigene Kalenderkategorie zum Tragen kommt.

Schlussendlich muss nun noch der Minikalender in der rechten Spalte eingebunden werden. Dieser wird im Gegensatz zu den Listen- und Detailansichten nicht über das

Backend eingepflegt. Dies wäre prinzipiell möglich, aber da der Minikalender auf jeder einzelnen Seite, die existiert, angezeigt werden soll, nicht sehr praktikabel. Von daher wird er mittels TypoScript direkt im TYPO3 Template verankert und standardmäßig ausgegeben.

Dem Marker für die rechte Spalte wird im TYPO3 Template das Objekt „right“ zugewiesen:

```
marks {
    CONTENT < content
    RIGHT < right
}
```

Das Objekt „right“ vom Typ „Content Objekt Array“ (COA) ist ein Sammelobjekt, welchem wiederum viele Unterobjekte unterschiedlichsten Typs zugewiesen werden können. In diesem Fall an der Stelle 10 die Definitionen für den Minikalender:

```
right = COA
right.10 < lib.minical
```

Die Definitionen für den Minikalender werden ebenfalls in TypoScript verfasst. Das Objekt „lib.minical“ ist hierbei wiederum vom Typ COA und enthält ein Unterobjekt vom Typ Text, welches die Überschrift bildet, und als nächstes Objekt das Kalender-Plug-in mit diversen für diese Ansicht notwendigen Einstellungen. Die Werte in geschweiften Klammern sind hierbei Konstanten, die in eine andere Datei ausgegliedert wurden, um sie schnell und einfach zu ändern und anzupassen:

```
lib.minical >
lib.minical = COA
lib.minical {
    5 = TEXT
    5.value = <h1 style="text-align:center;">Termine</h1>

    10 < plugin.tx_cal_controller
    10 {
        # Ausgangspunkte für alle Kalender relevanten
        # Punkte
```

```

pidList = {$pidList}
# das Plug-in wird nicht von einem extra Div
# umrandet
noWrapInBaseClass = 1
view {
    # die hier erlaubte Ansicht - nur
    # Monatsansicht ist erlaubt
    allowedViews = month
    # die zu verwendende HTML Designvorlage
    month.monthTemplate =
fileadmin/template/html/monthMini.txt
    # die ID der Seite, wo die Listenansicht
    # gerendert werden soll (es muss auf dieser
    # Seite auch das Plug-in vorhanden sein, um
    # die Liste zu rendern)
    list.listViewPid = {$showListId}
}
}
}

```

3.5.2. Volltextsuche

Die Volltextsuche soll sowohl in der Webpräsenz wie auch im Intranet zum Einsatz kommen.

3.5.2.1. Anforderungen

Die Anforderungen an die Suche sind vollständige und schnelle Suchergebnisse auf der Seite zu präsentieren. Die Suche soll sich dabei möglichst auf den Contentbereich beschränken und ressourcenarm arbeiten.

3.5.2.2. Variantenanalyse

Die Auswahl an geeigneten Such-Extensions ist sehr eingeschränkt. Dem zugrunde liegt, dass die mit am häufigsten genutzte Extension mittlerweile eine Systemextension ist. Dies bedeutet, sie wird voll von TYPO3 unterstützt und bereits im Installationspaket des WCMS mitgeliefert.

Indexed Search Engine (indexed_search)⁴⁸

Dies ist die Systemextension, welche überwiegend in den meisten TYPO3 Projekten verwendet wird.

Sie beruht auf zwei grundlegenden Techniken.

Erstens das Indizieren

Dies geschieht, während die Seite für das Frontend gerendert wird. Dabei werden alle Wörter auf dieser Seite registriert (den Bereich, in welchem die Wörter indiziert werden, kann man bestimmen), gezählt, gewichtet und anschließend in eine Datenbank-Tabelle geschrieben. Weiterhin wird eine zweite Tabelle mit Werten gefüllt, welche die Beziehungen zwischen den Wörtern und der aktuellen Seite herstellen.

Und zweitens das Suchen

Das Such-Plug-in, welches auf einer Seite eingebunden wird, durchsucht die erste Tabelle nach dem Auftreten des Such-Wortes und gibt anschließend alle Seiten aus, die damit in Verbindung stehen (zweite Tabelle) - die Sortierung berücksichtigt hierbei, wo auf der Seite das Suchwort zu finden ist, beziehungsweise wie häufig es auftritt.

Vorteil:

Es ist eine Systemextension und wird daher von TYPO3 komplett unterstützt. Es ist möglich, auch Dokumente wie PDF's in die Suche mit einzubinden. Und die Suche funktioniert zuverlässig.

Nachteil:

Es wird immer wieder von Performance-Problemen dieser Extension gesprochen. Ab einer gewissen Seitengröße mit vielen Inhalten bricht sie spürbar ein und verursacht Verzögerungen.

⁴⁸ o.V. - Indexed Search Documentation and Reference - http://typo3.org/extensions/repository/view/doc_indexed_search/current/ - Stand 10. August 2009

Alternativen

Mögliche Alternativen sind nur wenige vorhanden. Um den Performanceeinbrüchen bei sehr großen Seiten entgegen zu wirken, werden einige Extensions entwickelt, die auf der Lucene-Engine beruhen. Diese ist bei großen Datenmengen spürbar schneller⁴⁹. Als Nachteil bei dieser Suche wird der zwingende Einsatz von Java bezeichnet⁵⁰. Auch ist sie noch nicht sehr weit verbreitet und Support somit kaum vorhanden.

3.5.2.3. Entscheidung

Die Entscheidung fiel in dieser Arbeit auf die Verwendung der Indexed Search Engine. Einerseits sind die Vorteile durch die einfache Installation und den Support kaum zu überbieten. Und andererseits sind die Nachteile der Indexed Search Engine bei einer Projektgröße, wie es die Webpräsenz oder das Intranet darstellt, nicht wahrnehmbar, da kein Datenvolumen in dem Maße vorhanden ist, welches die starken Performanceeinbrüche hervorrufen könnte.

3.5.2.4. Konfiguration

Zum Einrichten der Indexed Search Engine benötigt es nicht viel. In erster Linie muss eine Seite angelegt werden, auf welcher das Plug-in eingebunden werden kann und wo letztendlich die Ergebnisse der Suche ausgegeben werden.

Des Weiteren muss auf allen Seiten im Frontend der Bereich gekennzeichnet werden, wo eine Indizierung von Daten stattfinden soll. Dadurch wird erreicht, dass zum Beispiel die Navigation als Solche nicht mit indiziert wird - was zur Folge hätte, dass ein Navigationstitel auf jeder Seite als Wort zählen würde, und damit das Ergebnis der Suche verfälschen würde.

⁴⁹ o.V. - TYPO3: Lucene vs. Indexed Search - Ein Performancetest - <http://www.form4.de/blog-news/blog/TYPO3-lucene-vs-indexed-search-ein-performancetest/> - Stand 16. August 2009

⁵⁰ o.V. - T3AK09 – SOLR Suche – Lucene based Search Engine -<http://TYPO3blogger.de/t3ak09-solr-suche-lucene-based-search-engine/> - Stand 16. August 2009

Um dies zu erreichen, wird in dieser Arbeit nur der Contentbereich zur Indizierung freigegeben. Der für diesen Bereich gültige Marker wurde bereits im TYPO3 Template mit den Inhalten verknüpft, welche die Redakteure im Backend anlegen können. Dieses Objekt wird nun mit den nötigen Angaben „gewrapt“ - umschlossen - und zum Indizieren freigegeben.

```
marks {  
    CONTENT < content  
    RIGHT < right  
}
```

```
content < styles.content.get  
content {  
    wrap = <!--TYPO3SEARCH_begin-->|<!--TYPO3SEARCH_end-->  
}
```

Um die Konfiguration der Indexed Search Engine abzuschließen, müssen mittels TypoScript im TYPO3 Template wieder Anpassungen vorgenommen werden, um die Suche den Wünschen nach entsprechend zu gestalten.

Das TypoScript wurde erneut ausgelagert, um es schnell und einfach mittels eines Editors bearbeiten zu können.

Der hierbei wichtigste Punkt, um die Such-Ergebnisse anpassen zu können, ist die Vorgabe der HTML-Designvorlage für dieses Plug-in:

```
plugin.tx_indexedsearch.Templatefile =  
fileadmin/kkh/template/html/indexed_search.html
```

Letztendlich muss noch das CSS angepasst werden, um die Suchseite abzuschließen.

Die Funktionalität ist somit gewährleistet und die Konfiguration abgeschlossen.

Durch das Pflichtenheft (beziehungsweise die Designvorlagen) ist jedoch vorgegeben, dass sich ein Inputfeld für die Suche auf allen Seiten der Webpräsenz befinden muss.

Die Indexed Search Engine ist dafür ungeeignet. Es gibt jedoch eine weitere Extension, um die Indexed Search Engine in diesem Punkt zu erweitern.

Die „Searchbox for Indexed Search Engine“ (macina_searchbox) wird ebenfalls mittels TypeScript konfiguriert (Zuweisung der zu verwendenden HTML-Designvorlage und der PageID der Seite, wo das Such-Plug-in eingebunden ist) und anschließend einem Marker zugewiesen.

```
plugin.tx_macinasearchbox_pi1 {  
    pidSearchpage = {$searchpage}  
    templateFile =  
    fileadmin/kkh/template/html/macina.html  
}
```

```
searchbox < plugin.tx_macinasearchbox_pi1
```

```
marks {  
    SEARCHBOX < searchbox  
}
```

3.5.3. News

Eine Extension zur Verwaltung von News kommt derzeit nur im Intranet zum Einsatz. Sie soll genutzt werden, um Neuigkeiten in den einzelnen Bereichen schnell und einfach veröffentlichen zu können.

3.5.3.1. Anforderungen

Die Mindestanforderungen an eine Extension zur Darstellung von Neuigkeiten sind folgende:

- Einfache Pflege der News über das Backend
- eine Listen- und Detailansicht aller Neuigkeiten
- Kategorisierung von News - dieser Punkt ist wie bereits bei der Kalenderextension von Bedeutung, um den Anforderungen des Intranets gerecht zu werden

3.5.3.2. Variantenanalyse

Die Recherche nach einer möglichen Extension hatte nur die Erweiterung tt_news zum Ergebnis. Diese mittlerweile sehr ausgereifte Extension erfüllt alle Voraussetzungen. Sie bietet eine Vielzahl verschiedener Ansichten, unterstützt das Arbeiten mit Kategorien und ist sehr anpassungsfähig. Weiterhin sind die Dokumentation und die im Internet zu findenden Hilfestellungen sehr umfangreich.

3.5.3.3. Konfiguration

Für den Betrieb der Extension werden, wie bereits bei der Kalender Erweiterung, Seiten für die Ansichten Liste und Single benötigt. Diese werden als erstes angelegt und folgend die Plug-ins für die News eingebunden.

Anschließend wird erneut das TYPO3 Template um Definitionen für die News Extension erweitert. Unter anderem wird erneut die HTML-Designvorlage definiert, sowie der „Startpunkt“ für das Plug-in, ab wo es nach angelegten News-Datensätzen suchen kann. Diese PageID verweist in dieser Arbeit wieder auf einen Systemordner in der Ordnerstruktur parallel zum Seitenbaum (wie beim Kalender auch).

3.5.4. Weitere Extensions

Weitere in dieser Arbeit verwendete Extensions, deren Auswirkung auf die Seite jedoch nicht so umfangreich ist, sind unter anderem:

3.5.4.1. Flash Page Header (flash_pageheader)⁵¹

Zielsetzung

Ziel ist es, für jede Seite im Frontend der Webpräsenz individuell sowie schnell und einfach einen anderen Flashfilm integrieren zu können. Voraussetzung hierfür ist, dass alle Flash Filme die selben vorgegebenen Maße erfüllen (Breite und Höhe).

Umsetzung

Umgesetzt wird dies durch die Verwendung der Flash Page Header Extension.

Das Plug-in wird mittels TypoScript dem passenden Marker zugeführt

```
flash < plugin.tx_flashpageheader_pi1
```

```
marks {  
    FLASH < flash  
}
```

Konfiguriert wird dieses Plug-in ebenfalls mittels TypoScript. Unter anderem werden Werte wie „defaultswf“ oder allgemeine Flashfilm Einstellungen vorgenommen:

- *plugin.tx_flashpageheader_pi1.width* - Breite des Films
- *plugin.tx_flashpageheader_pi1.height* - Höhe des Films
- *plugin.tx_flashpageheader_pi1.defaultSwf* - der Defaultflashfilm welcher abzuspielen ist, falls keine Alternative angegeben wurde
- *plugin.tx_flashpageheader_pi1.swfParams.loop* - Flash Parameter - gibt an, ob der Flashfilm wiederholt werden soll
- *plugin.tx_flashpageheader_pi1.swfParams.quality* - Flash Parameter - gibt die zu verwendende Animationsqualität des Flashfilms an

⁵¹ o.V. - Flash Page Header - http://typo3.org/extensions/repository/view/flash_pageheader/current/ - Stand 18. August 2009

Mit den bisherigen Einstellungen wird auf allen Seiten nun der als Default eingestellte Flashfilm angezeigt. Mit der Installation steht im Backend allerdings ein neues Inputfeld zur Verfügung. Beim Bearbeiten einer Seite ist dieses unter dem Punkt „Erweitert“ erreichbar. Das Einstellen eines individuellen Flash Films für jede Seite ist hier möglich.

3.5.4.2. Powermail (powermail)⁵²

Zielsetzung

Es soll ein Kontaktformular erstellt werden, über das Nutzer der Webpräsenz Kontakt zur Klinikum Schönebeck GmbH herstellen können.

Umsetzung

Für die Umsetzung wird auf die Extension Powermail zurückgegriffen. Sie bietet umfangreiche Möglichkeiten zum Erstellen und Konfigurieren komplexer Formulare.

Auf der Seite, wo das Kontaktformular erscheinen soll, wurde das Plug-in integriert. Hierdurch können die benötigten Inputfelder generiert werden. Es unterstützt auch die Möglichkeit, einzelne Felder als Pflichtfeld zu definieren, was bei Benutzung mittels Javascript überprüft wird. Auch eine Überprüfung der eingegebenen Daten nach regulären Ausdrücken mittels PHP und Javascript ist möglich. Dadurch wird überprüft, ob eine eingetragene Email auch tatsächlich eine gültige Email-Adresse sein könnte.

Weiterhin wird eingestellt, an welche Email-Adresse die eingetragenen Informationen gesandt werden soll.

Die Formatierung des Designs erfolgt mittels CSS.

Die Texte, welche in der Email noch enthalten sind, können über das Backend editiert werden.

⁵² o.V. - powermail - <http://typo3.org/extensions/repository/view/powermail/current/> - Stand 16. August 2009

3.5.4.3. Tip-A-Friend Plus (tipafriend_plus)⁵³

Zielsetzung

Dem User soll eine Möglichkeit gegeben werden, eine Empfehlung für eine bestimmte Seite an eine ihm bekannte Person zu senden.

Das Verfahren soll für den User einfach gehalten werden.

Diese Möglichkeit wird nur auf der Webpräsenz geschaffen.

Umsetzung

Für die Umsetzung dieses Ziels wurde die Extension Tip-A-Friend Plus genutzt. Diese unterstützt die geforderten Anforderungen. Dem User wird dabei auf jeder Seite im Frontend ein Link zur Verfügung gestellt, welcher diesen zu einem einfachen Formular weiter leitet. In diesem ist die Email-Adresse des Ziels und des Absenders zu hinterlegen, sowie ein kleiner Text, welcher der Email an den Bekannten beigelegt wird.

Um diese Funktionalität umzusetzen, musste eine neue Seite erstellt werden, in welcher das Plug-in zum Weiterempfehlen eingebunden werden kann. An dieser Stelle wird mittels des Plug-in das Formular zur Verfügung gestellt. Um den Link zum Weiterempfehlen dem Nutzer der Webseite auf allen Seiten zugänglich zu machen, musste erneut mittels TypoScript das Plug-in einem Marker zugewiesen werden. Die Formatierung des Formulars geschieht mittels CSS sowie durch den Einsatz eines HTML-Templates, welche ebenfalls über TypoScript dem Plug-in zugeordnet wird.

3.6. Closed User Groups / Rechtevergabe

Die Möglichkeit der Vergabe von Rechten ist ein grundlegender Bestandteil eines WCMS und wird in der Regel dafür genutzt, Nutzern des Systems nur soweit eingeschränkten Zugang zu geben, dass sie die an sie gestellten Aufgaben erfüllen

⁵³ o.V. - Tip-A-Friend Plus - http://typo3.org/extensions/repository/view/tipafriend_plus/current/ - Stand 17. August 2009

können, jedoch zu sensiblen Bereichen, die für die Funktionalität der Seite sehr wichtig sind, keinen oder nur beschränkten Zugang erhalten.

TYPO3 ist weiterhin ein Mehrbenutzersystem. Dies bedeutet, dass mehrere Benutzer durchaus gleichzeitig mit dem System arbeiten können

3.6.1. Gruppen

TYPO3 erlaubt die Arbeit mit Gruppen. So ist es möglich, User mit selben oder ähnlichen Aufgabenbereichen in ihnen zusammenzufassen. Die Vergabe von Zugriffsrechten sowie die Zuordnung von Systemressourcen wie zum Beispiel Datenbank- und Verzeichnisfreigaben müssen somit nicht mehr individuell für jeden Nutzer geschehen.

Der Aufbau der Gruppenstruktur ist hierbei baumähnlich. Jeder Benutzer ist dabei das Wurzel- und somit höchste Element. Ihm zugewiesen werden die Gruppen. Dabei kann ein Nutzer durchaus mehrere Gruppen besitzen. Auch ist es möglich, dass Gruppen selbst wiederum andere Gruppen als Kindelemente zugewiesen bekommen.

Die Vererbung der Rechte folgt anschließend von den Blättern des Baums hin zu seiner Wurzel. Ein User mit zugewiesener Gruppe erbt also die Rechte, welcher der Gruppe zugeordnet sind inklusive jener Rechte, welche die Gruppe selbst von anderen Gruppen vererbt bekommt.

Durch diese Struktur sind zahlreiche Konstrukte möglich, welche die Rechtevergabe aller User bestimmen.

3.6.2. Webpräsenz

Ziel ist es, der Klinikum Schönebeck GmbH eine selbstständige Pflege der Inhalte zu ermöglichen.

Umgesetzt wird dies durch Schaffung mehrerer Backendzugänge mit ausreichenden Rechten zur redaktionellen Arbeit.

Im konkreten Fall der Webpräsenz sollen zwei Backendzugänge für diese Arbeit geschaffen werden. Eine Erweiterbarkeit um weitere Backendzugänge mit selber Rechtevergabe soll leicht und schnell durchführbar sein.

Um das zuletzt genannte Ziel zu erreichen, wird die genaue Definition der Rechte in eine Gruppe ausgelagert.

Diese Gruppe wird anschließend den neuen Usern zugewiesen, so dass diese die Rechte von ihr erben.

Gruppen wie auch User selbst sind in TYPO3 Datensätze, welche über das Wurzelement des Seitenbaums verwaltet werden. Über den Menüpunkt „neuer Datensatz“ innerhalb des Wurzelementes ist es so möglich, eine neue Gruppe anzulegen.

Zur Konfiguration des Datensatzes stehen 4 Reiter zur Verfügung:

- Allgemein
- Zugriffsliste
- Freigaben und Arbeitsumgebung
- Optionen

Innerhalb des ersten Punktes wird der Name definiert - in diesem Fall „Redakteure Internet“ - sowie eine kurze Beschreibung, wofür diese Gruppe genutzt wird.

Außerdem kann man innerhalb dieses Reiters weitere Untergruppen dieser Gruppe hinzufügen. Für die Webpräsenz ist jedoch die Verwendung einer einzigen Gruppe völlig ausreichend.

Unter dem zweiten Reiter „Zugriffsliste“ findet die eigentliche Bestimmung der Rechte statt. Bei einer neu angelegten Gruppe oder auch beim Anlegen eines Users sind alle Felder vorerst leer. Es sind also keine Rechte zugewiesen.

Der erste Punkt in diesem Reiter sind die Module. Hier bestimmt man, welche Module in der Hauptnavigation des Backends dem User angezeigt werden.

Die Redakteure bekommen die Zugänge für das Modul „Web“ inklusive der Unterpunkte „Seite“ und „Liste“, für das Modul „Datei“ sowie des Unterpunktes „Dateiliste“, für das Modul „Benutzerwerkzeuge“ inklusive des Unterpunktes „Benutzereinstellungen“ und für das Modul „Hilfe“. Zugang zu allen anderen Modulen wird nicht gewährt.

Die nächsten Punkte im Reiter Zugriffsliste sind die Tabellen. Hier wird einerseits angegeben, welche Tabellen der User sehen darf und welche er auch ändern darf. Der Begriff Tabelle ist hierbei im ersten Moment etwas verwirrend. Im Prinzip geht es darum, welche Datensätze der User sehen beziehungsweise ändern darf.

In diesem Fall bekommen die Redakteure der Webseite Zugang zu den Tabellen „Seite“ und „Seiteninhalt“. Damit sind bereits nahezu alle Datensätze abgedeckt, welche die Redakteure einsehen und ändern dürfen. Hinzu kommen noch Tabellen für die Formularelemente und Adressdatensätze.

Um die Redakteure in der Auswahl einzuschränken, welche Seitentypen sie anlegen können, wurden ihre Rechte im nächsten Punkt auf die folgenden Typen beschränkt: „Standart“, „Verweis“, „Externer Link“ und „SysOrdner“.

Der größte und umfangreichste Punkt befasst sich mit der Auswahl der Felder, welche den Usern zum Bearbeiten freigegeben werden. Hier sind die meisten Felder gelistet, die im Backend vorkommen können. Der Zugang wird zu all jenen gewährt, welche für das redaktionelle Arbeiten wichtig sind. In erster Linie sind dies die Felder im Modul Web für Seiten und Seiteninhalte, sowie zur Angabe von Adressdatensätzen.

Der vorerst letzte Punkt innerhalb dieses Reiters legt fest, welchen Typs neu anzulegende Inhaltselement sein dürfen, beziehungsweise welche Typen explizit nicht erlaubt sind. So werden die Standarttypen wie Text oder Text mit Bild erlaubt, andere Typen wie zum Beispiel die Sitemap verboten.

Es konnte jetzt definiert werden, welche Inhalte der User sehen kann und welche nicht. Das nächste Ziel ist zu bestimmen, zu welchem Teil des kompletten Seitenbaums der User Zugriff hat.

Im dritten Reiter gibt es hierfür zwei Punkte:

- Datenbankfreigabe
- Verzeichnisfreigabe

Der Erste bezieht sich direkt auf den Seitenbaum. In diesem Fall wurde Zugriff auf das höchste Seitenelement der Webseite (die „Wurzel“-Seite), und zweitens auf das höchste Element der parallel dazu angeordneten Sysordner-Struktur. Somit können die Redakteure den kompletten Seitenbaum einsehen, sowie alle Datensätze in der Sysordner-Struktur benutzen.

Der Zweite bezieht sich auf die Ordnerstruktur direkt auf dem Server. Hier werden physisch vorhandene Daten wie Bilder, PDF's oder Filme gespeichert.

Der Zugriff auf die Weltkugel - das aller höchste Element einer jeden TYPO3 Seite - wird verwährt. Die hier verwalteten Datensätze wie User oder Usergruppen werden von Redakteuren nicht benötigt.

Unter dem vierten und letzten Reiter „Optionen“ können noch weitere Definitionen zum Beispiel mittels TypoScript getätigt werden, was bei der Verwaltung der Rechte in dieser Webpräsenz jedoch nicht von Nöten ist.

Damit ist die Bearbeitung der Gruppe abgeschlossen.

Um den zwei Redakteuren der Klinikum Schönebeck GmbH nun Zugang zu gewähren, müssen zwei User Datensätze angelegt werden.

Diese werden ebenfalls auf oberster Ebene unter der Weltkugel verwaltet, und sind ähnlich aufgebaut wie auch die Gruppendatensätze. Unter dem Reiter „Allgemein“ werden hier jedoch noch Login und Passwort angegeben.

Unter dem Punkt Gruppen wird nun die zuvor erstellte Gruppe „Redakteure Internet“ ausgewählt und hinzugefügt. Hierdurch erben die User die der Gruppe zugewiesenen Rechte.

Der nächste Reiter „Zugriffsrechte“ bleibt unangetastet, da alle benötigten Einstellungen über die Gruppe bereits geregelt wurden. Hier wäre es aber möglich, dem einzelnen User individuell erweiterte Rechte zu gewähren.

Im dritten Reiter „Freigaben und Arbeitsumgebungen“ werden ebenfalls kaum Einstellungen vorgenommen. Es wird durch Setzen eines Hakens nur bestimmt, dass die Datenbank- und Verzeichnisfreigaben der zugewiesenen Gruppe auch übernommen werden soll.

Im 4. Reiter ist die Angabe von TypoScript wieder möglich, um die Rechte des Users noch weiter zu bearbeiten. Wie auch bei der Gruppe wird dies jedoch unangetastet gelassen.

Der 5. Reiter, welcher bei den User Datensätzen noch zusätzlich vorhanden ist, bleibt ebenfalls unangetastet. Über diesen ist es möglich, diesen Useraccount zeitlich zu begrenzen. Durch Angabe von Start- und Endzeitpunkt wird ein Zeitraum definiert, in welcher der Account Gültigkeit besitzt.

Nach Anlegen der beiden Datensätze stehen nun der Klinikum Schönebeck GmbH zwei Zugänge zum redaktionellen Arbeiten zur Verfügung.

3.6.3. Intranet

Die Ansprüche an die Rechtestruktur des Intranets sind ungleich umfangreicher. Grundlegend ist hierbei auch auf die Unterscheidung von Frontend und Backend zu achten. Beide Zugänge werden von TYPO3 unterschiedlich behandelt und getrennt verwaltet.

Als erstes werden nachfolgend die nötigen Einstellungen für die Backend User behandelt.

3.6.3.1. Rechte - Backend

3.6.3.1.1. Zielsetzung

Die grundlegenden Ziele sind folgende:

- individuelle Backendzugänge für jede Fachabteilung. Diese sollen dabei auf ihre Fachabteilung beschränkt sein.
- Fachabteilung übergreifende Backendzugänge.
- unterschiedliche Rechte (zum Beispiel Zugriff auf Userverwaltung, ja oder nein)
- Fachbereichsredakteure sollen News und Termine nur für ihren Fachbereich anlegen dürfen. Übergeordnete Redakteure und Administratoren sollen die Möglichkeit erhalten, diese News und Termin auch in anderen Fachbereichen zu veröffentlichen. Es soll hierbei auf die Verwendung von Kopien verzichtet werden, um Dateninkonsistenz zu vermeiden.
- Die Userverwaltung soll langfristig von der Klinikum Schönebeck GmbH selbstständig durchgeführt werden. Hierbei ist darauf zu achten, die komplexen Möglichkeiten der Verwaltung dem hierfür zuständigen zukünftigen Administrator so einfach wie möglich zu gestalten.

3.6.3.1.2. Umsetzung

Um die Anforderungen zu erreichen, wird wieder auf die Verwendung von Usergruppen zurückgegriffen.

Den Zielen entsprechend wird versucht, User mit Gemeinsamkeiten in ihrer Arbeit zu gruppieren. Dabei stellen sich pro User zwei unterschiedliche Aspekte heraus. Einerseits, wo die User zum Einsatz kommen - also die Beschränkung auf einen oder mehrere Fachbereiche - und andererseits die Rechte, was ein User tun darf - zum Beispiel redaktionelle Arbeit.

Hieraus folgernd kann jeder User durch Zuweisung jeweils einer Gruppe für die Rechte und einer Gruppe für den „Arbeitsort“ bestimmt werden.

3.6.3.1.2.1. Rechte

Wie in Kapitel 3.6.2 Webpräsenz beschrieben, werden mehrere Usergruppen im Backend des Intranets angelegt. Um die Arbeit des zukünftigen User-Administrators der Klinikum Schönebeck GmbH zu erleichtern, wird hierbei auf eine eingängige und eindeutige Namensgebung geachtet. Die ersten beiden erstellten Gruppen beziehen sich auf die möglichen Rechte der User. Diese erhalten den Namen „Rechte:Redakteure“ und „Rechte:Chefredakteure“. Bei der Erstellung beider Gruppen werden nur die Reiter „Allgemein“ und „Zugriffsleiste“ bearbeitet.

3.6.3.1.2.2. Einstiegspunkt

Die nächsten erstellten Gruppen gewähren keinerlei Rechte sondern bestimmen nur, welche Teile des Seitenbaums für die User sichtbar sind. Hierfür werden Einstellungen im Reiter „Freigaben und Arbeitsumgebungen“ vorgenommen. Es wird für jede Fachabteilung eine eigene Gruppe angelegt mit Definitionen für die Einstiegspunkte in die Seitenbaum- sowie die Ordnerstruktur, so dass Mitglieder dieser Gruppe nur in diese Bereiche Einblick erhalten. Durch die Verwendung der Extensions „tt_news“ (News) und „cal“ (Terminkalender) ist der Datensatz Usergruppe weiterhin um einen fünften Reiter ergänzt. In diesem Reiter „Erweitert“ stehen zwei weitere Auswahlmöglichkeiten zur Verfügung. Zuvor angelegte Kategorien für News und Termine können hier ausgewählt und somit für Nutzer dieser Usergruppe erlaubt werden. Dadurch ist es auch möglich, dass ein Redakteur mit Zugängen zu mehreren Bereichen bereits bestehende oder neue News um weitere Kategorien ergänzt, und somit im Frontend auf unterschiedlichen Seiten veröffentlicht.

Die einheitliche Namensgebung bei der Gruppendifinition wird hier fortgesetzt. Ein vorangestelltes „SeitenEinstieg:“ gefolgt vom Namen der Fachabteilung sorgt für ein leichtes Erkennen der Gruppe.

Durch Zuweisen mehrerer „Einstiegsgruppen“ ist es somit auch möglich, einem User Zugang zu mehreren Fachabteilungen zu gewähren.

Für einen User mit zum Beispiel Administratorrechten, welcher Zugang zu allen Bereichen der Seite erhalten soll (nicht nur Fachabteilungen) wurde eine weitere

„Einstiegsgruppe“ angelegt, mit Einstiegspunkt auf dem Wurzelementes des Seitenbaums und Zugriff auf alle Kategorien der News- und Kalenderextension (Name der Gruppe: „SeitenEinstieg:all“) . Somit erhält er auch Zugriff zu Teilen, die keinem Fachbereich zugeordnet sind.

3.6.3.1.2.3. Besitz

Ein weiterer wichtiger Punkt bei der Vergabe von Rechten und Zugriffen ist der „Besitz“ von Seiten.

Über das Modul „Web“ ist der Punkt „Zugriff“ zu erreichen. Über diesen kann der „Besitz“ und die Gruppenzugehörigkeit von Seiten eingesehen und verändert werden, beziehungsweise die fünf Rechte an dieser Seite angepasst werden. Die Rechte sind folgende:

- Seite anzeigen - Seite und Inhalt anzeigen/kopieren
- Inhalt bearbeiten - Inhalt ändern/erstellen/löschen/verschieben
- Seite bearbeiten - Seite ändern/verschieben (z.B. Seitentitel ändern usw.)
- Seite löschen - Seite und Inhalt löschen
- Neue Seiten - Neue Seiten unter dieser Seite erstellen

Der Besitzer einer Seite (standardmäßig der User, welcher sie erstellt) darf alles. Die Gruppenmitglieder der Usergruppe, zu der die Seite gehört, dürfen in der Regel alles außer löschen. Allen Anderen sind keinerlei Rechte an dieser Seite gewährt.

Während bei der Webpräsenz der Besitz einer Seite nur eine untergeordnete Rolle spielt, da es nur eine Usergruppe gibt und somit alle dieser Gruppe zugehörigen User die benötigten Rechte haben um damit zu arbeiten, ist dies im Intranet nicht der Fall.

Redakteure mit Zugang zu nur einer Fachabteilung sollten immer mindestens die Gruppenrechte besitzen, um ihre Arbeiten bewerkstelligen zu können. Erstellt jedoch zum Beispiel ein Administrator mit anderer Gruppenzugehörigkeit die Seite, so verlieren die Redakteure alle Rechte und können die Seite nicht einmal einsehen.

Um dies zu verhindern ist es möglich, innerhalb des „Page TSconfig“ einer Seite (Kapitel 2.2.4.2 TypoScript) die Gruppenzugehörigkeit einer jeden Seite zu bestimmen.

Mittels folgendem TypoScript Code wird die Zugehörigkeit aller unterhalb dieser Seite angelegten Seiten festgelegt (in diesem Beispiel wird die Gruppenzugehörigkeit zur Usergruppe mit der ID 6 hergestellt). Diese Einstellung wird dabei an alle Unterseiten vererbt, bis sie durch eine neue Einstellung überschrieben wird.

```
### Rechtevergabe für neu erstellte Seiten ###
TCEMAIN {
    permissions.groupid = 6
}
```

Diese Einstellung muss einmal für die Wurzelseite des kompletten Baums, sowie für die Wurzelseiten der einzelnen Fachabteilungen getätigt, und jeweils die passende „Einstiegsgruppe“ (3.6.3.1.2.2) gewählt werden.

Jetzt können alle Redakteure die Seiten innerhalb ihrer Fachabteilung bearbeiten, unabhängig davon, wer die Seite erstellt hat.

Jedoch ergibt sich hieraus und der in Kapitel 3.6.3.1.2.2 erstellten Usergruppe „Einstiegspunkt:all“ für Administratoren ein Problem. Denn diese haben keinen Zugriff auf diese Seiten, weil sie keiner fachspezifischen „Einstiegsgruppe“ angehören.

Um dieses Problem zu lösen, wäre es möglich, dem User auch die anderen Einstiegsgruppen zuzuweisen. Jedoch wäre dies mit einem anderen Problem verbunden. Der User hätte zusätzlich zu den in seiner „SeitenEinstieg:all“ definierten Startpunkten die Einstiegspunkte aus den anderen Gruppen, und diese damit doppelt.

Letztendlich wurde das Problem durch Schaffung einer dritten Gruppenkategorie gelöst. Neben den Gruppen für Rechte und Einstiegspunkte gibt es jetzt weitere Gruppen für Besitz. Diese folgen der bisher verwendeten Namensgebung und

beginnen mit „Besitzer:“ gefolgt von dem Namen der Fachabteilung beziehungsweise „all“ für alle anderen Seiten.

Diesen Gruppen werden hierbei keinerlei Rechte vergeben. Sie symbolisieren nur den Besitz. Die in den „Page TSconfig“ gemachten Einstellungen werden dahingehend geändert, dass diese den „Besitzergruppen“ zugeordnet sind.

Das Ursprungsproblem, dass ein Administrator die Seiten der Fachabteilungen nicht einsehen kann, kann jetzt dadurch behoben werden, dass die ihm zugewiesene Gruppe „Besitzer:all“ alle anderen „Besitzergruppen“ als Untergruppen enthält, und somit das Besitzrecht von diesen erbt.

3.6.3.1.2.4. User

Die Erstellung individueller Benutzer gestaltet sich auf Grund des vorangegangenen Gruppenaufbaus sehr einfach. Zusätzlich zu den allgemeinen Angaben wie Name, Login und Passwort ist die Wahl der dem User zugewiesenen Gruppen entscheidend.

Hierbei ist darauf zu achten, jedem User mindestens jeweils eine Gruppe aus allen drei Kategorien zuzuweisen. Durch Zuweisung folgender drei Gruppen

- Rechte:Redakteur
- SeitenEinstieg:Chirurgie
- Besitzer:Chirurgie

wird ein Backendbenutzer geschaffen, welcher die Rechte eines Redakteurs innerhalb des Fachbereiches Chirurgie ausüben darf.

3.6.3.2. Rechte - Frontend

Die Rechte für das Frontend werden komplett getrennt von den Backendusern verwaltet. Es stehen auch hier Datensätze zum Anlegen von Usern sowie Usergruppen zur Verfügung. Die Verwendung von Usergruppen zur Strukturierung

der Zugänge wird auch im Frontend verwendet. Jedoch werden diese nicht innerhalb der Weltkugel verwaltet, sondern innerhalb des Seitenbaums.

3.6.3.2.1. Zielsetzung

- Zugang durch Login beschränkt
- allgemeine Informationen jedem Benutzer zugänglich
- interne Informationen der Fachabteilungen ihren jeweiligen Mitgliedern vorbehalten

3.6.3.2.2. Umsetzung

Für die Umsetzung eines passwortgeschützten Bereiches ist zunächst eine Seite zum Einloggen notwendig. Auf dieser wird das Login-Plug-in eingebunden. Hierdurch wird das Anmelden des FE-Users an die Seite ermöglicht. Um weiterhin den Zugang auf die Inhaltsseiten nicht angemeldeten FE-Users zu verweigern, müssen diese in Ihrem Zugriff beschränkt werden. Beim Bearbeiten einer Seite kann man unter dem Reiter „Zugriff“ bestimmen, welche FE-Usergruppe eine Seite einsehen darf. Werden hier keine Einstellungen vorgenommen, so ist die Seite standardmäßig für jedermann sichtbar.

Um einen allgemein zugänglichen Bereich für alle User zu kreieren, wird als erstes die Usergruppe „Frontendzugang“ geschaffen. Allen Seiten, welche von jedem eingeloggten FE-User besucht werden darf, wird diese Gruppe als Zugangsbeschränkung zugewiesen.

Weitere Gruppen zur Zugangsbeschränkung für interne Informationen innerhalb der Fachbereiche sind notwendig. Für jede Fachabteilung wird eine weitere FE-User Gruppe definiert. Seiten mit internen Informationen werden nun diese als Beschränkung gegeben. Weiterhin erben alle spezialisierten Gruppen die Gruppe „Frontendzugang“ als Untergruppe und erhalten somit ebenfalls Zugang zu allen allgemeinen Informationen.

Die anzulegenden FE-User erhalten anschließend eine Gruppe, über welche bestimmt wird, wie weit Ihr Zugang reicht.

3.7. Datensicherheit

Beim Thema Datensicherheit gibt es zwei grundsätzliche Punkte, die es zu beachten gilt.

Erstens der Schutz vor Datenverlust. Ursachen hierfür können technische Mängel, Fehlbedienung oder einfach höhere Macht wie zum Beispiel Feuer oder Blitzschlag sein.

Und zweitens der Schutz vor Fremdeinwirkung mit dem Ziel des Datendiebstahls oder der Datenmanipulation.

3.7.1. Schutz vor Datenverlust

Das wichtigste Mittel gegen den Verlust von Daten ist ein Backup⁵⁴. Hierbei müssen alle sicherungsrelevanten Daten kopiert werden. Um den höchstmöglichen Schutz zu gewähren, ist diese Datensicherung regelmäßig durchzuführen.

In diesem Fall wird dies durch den Hoster Mittwald täglich durchgeführt. Ein Wiederherstellen von verlorenen Daten ist somit möglich.

3.7.2. Schutz vor Fremdzugriff und Diebstahl oder Manipulation

Um Schutz vor Fremdzugriff zu gewährleisten, sollten als erstes die bereits bestehenden Zugriffsmöglichkeiten näher betrachtet werden.

Über das Backend haben je nach Projekt bereits unterschiedlichste Personen Zugang. TYPO3 protokolliert hierbei alle vorgenommenen Änderungen an Daten und

⁵⁴ Das Anfertigen von Kopien der zu sichernden Daten auf ein anderes (häufig transportables) Speichermedium.
- o.V. - Datensicherung - <http://de.wikipedia.org/wiki/Backup> - Stand 19. August 2009

Inhalten inklusive der Uhrzeit und der IP des Users⁵⁵. Um generell Missbrauch vorzubeugen ist jedoch eine Einschränkung der Rechte eines jeden User auf das Notwendigste nötig. In dieser Arbeit wird dies durch eine komplexe Userverwaltung bewerkstelligt (Kapitel 3.6).

Weitere Angriffspunkte können durch Sicherheitslücken, welche zum Beispiel SQL-Injections⁵⁶ ermöglichen, gefunden werden. Jedoch wird der Kern des WCMS TYPO3 als sehr sicher eingestuft⁵⁷. Weiterhin werden gefundene Sicherheitslücken durch das Bereitstellen von „Securitypatches“ schnell behoben. Durch Verwendung des TYPO3 Hoster Mittwald wird man über das Erscheinen zwingend notwendiger Patches schnell und zuverlässig informiert.

Häufiger auftreten können Sicherheitslücken in Extensions, da diese von Dritten programmiert werden und eventuell Fehler enthalten. Das Risiko minimieren kann man, wenn man sich bei der Auswahl der zu verwendenden Erweiterungen einschränkt. Die im Erweiterungsmanager angezeigten Zahlen bezüglich der Downloads einer Extension sagen oft bereits viel aus. Im Allgemeinen kann man feststellen, dass je häufiger eine Extension herunter geladen und damit genutzt wurde, desto ausgereifter und sicherer ist sie⁵⁸. „Reviewed“ Extensions bieten die größte Sicherheit. Diese Erweiterungen sind von einem Sicherheitsteam von typo3.org getestet⁵⁹, und man kann von hoher Sicherheit ausgehen.

Absolute Sicherheit ist nicht zu erreichen. Aber durch einen verantwortungsvollen Umgang mit den Backendzugangsmöglichkeiten, zeitnaher Einspielung sicherheitsrelevanter Updates sowie der Datensicherung kann man die Risiken minimieren.

⁵⁵ o.V. - Sicherheit eines TYPO3 Content Management Systems - <http://www.aixwave.com/leistungen-produkte/cms-systeme-typo3/typo3-sicherheit.html> - Stand 19. August 2009

⁵⁶ „SQL-Injection (dt. SQL-Einschleusung) bezeichnet das Ausnutzen einer Sicherheitslücke in Zusammenhang mit SQL-Datenbanken, die durch mangelnde Maskierung oder Überprüfung von Metazeichen in Benutzereingaben entsteht. Der Angreifer versucht dabei, über die Anwendung, die den Zugriff auf die Datenbank bereitstellt, eigene Datenbankbefehle einzuschleusen. Sein Ziel ist es, Daten in seinem Sinne zu verändern oder Kontrolle über den Server zu erhalten.“ - o.V. - SQL-Injection - <http://de.wikipedia.org/wiki/SQL-Injection> - Stand 19. August 2009

⁵⁷ Jens Tißler - Interview zur TYPO3-Sicherheit - <http://t3n.yeebase.com/interview-typo3-sicherheit-denkt-keine-fehler-machen-243531/> - Stand 17. August 2009

⁵⁸ o.V. - ist TYPO3 sicher - <http://typo3.fruit-lab.de/typo3-sicherheit.html> - Stand 18. August 2009

⁵⁹ o.V. - ist TYPO3 sicher - <http://typo3.fruit-lab.de/typo3-sicherheit.html> - Stand 18. August 2009

4. Zusammenfassung und Ergebnisse

Innerhalb dieser Arbeit wurden eine neue Webpräsenz sowie ein Intranet geschaffen. Beide sind grafisch wie technisch auf der Höhe der Zeit. Die Anforderungen an beide Systeme wurden erfüllt. Durch Verwendung des WCMS TYPO3 ist die Pflege durch die Klinikum Schönebeck GmbH möglich.

Zum aktuellen Zeitpunkt ist die Webpräsenz bereits online und hat die alte Seite ersetzt. Und es wurden einige weitere Anpassungen an der Seite durchgeführt, unter anderem die Möglichkeit der Einbindung von Flashfilmen im Contentbereich der Seite oder die Einbindung von Googlemaps zum Zweck der Anfahrtswegkennzeichnung.

Das Intranet ist noch nicht zum endgültigen Abschluss gekommen. Einige offene Punkte bezüglich der Userverwaltung zum Beispiel müssen noch zwischen der mediatack GmbH und der Klinikum Schönebeck GmbH geklärt werden.

Unter folgenden Links können beide Projekte erreicht werden:

- <http://www.kreiskrankenhaus-schoenebeck.de/>
Stand 20. August 2009
- <http://p105325.typo3server.info/>
Stand 20. August 2009

5. Ausblick

Die Webpräsenz ist bereits online. Kleinere Erweiterungen, welche den Funktionsumfang des Systems erweitern, können jederzeit folgen.

Die Primäre Aufgabe für die Zukunft ist jedoch die Fertigstellung des Intranets. Nach Klärung der offenen Punkte ist es das Ziel, dieses ebenfalls fertig zu stellen und es der Pflege durch die Klinikum Schönebeck GmbH zu unterstellen.

6. Anhang

Nachfolgende sollen einige Codebeispiele vorgestellt werden. Diese sind überwiegend HTML-Designvorlagen oder TypoScript-Anweisungen.

6.1. HTML-Templates

Nachfolgend werden einige der erstellten Designvorlagen vorgestellt. Überwiegend sind dies die für die Grundstruktur der Seite notwendigen Main- und Sub HTML-Templates am Beispiel der Webpräsenz. Die zugrunde liegende Idee und der Aufbau findet auch bei den HTML-Designvorlagen des Intranet Verwendung.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Unbenanntes Dokument</title>
<link rel="stylesheet" media="all" type="text/css"
href="kkh.css" />
</head>

<body>

<!-- ###DOCUMENT### -->
<div id="container">
  <div id="header_01">
    <div id="headerlink">###HEADERLINK###</div>
    ###SEARCHBOX### ###SERVICENAVIGATION###
  </div>
  <div id="header_02">
    <div id="flash">
      ###FLASH###
    </div>
    <div id="navi">
      ###NAVIGATION###
    </div>
    <div class="clear"></div>
  </div>
  <div id="line_01"></div>
  ###SUBTEMPLATE###
```

```

</div>
<div id="line_02"></div>
<div id="footer">&nbsp;###FOOTER###</div>
<!-- ###DOCUMENT### -->

</body>
</html>

```

Anhang 1 - main.html - Main Template. Der für das WCMS notwendige Bereich, wird durch die Marker `###DOCUMENT###` eingegrenzt. jeglicher HTML Code außerhalb dieses Bereiches hat keinerlei Auswirkung und wird komplett ignoriert. Für die Entwicklung der Vorlage ist er jedoch vorteilhaft, weil hierdurch diese ohne Verwendung von TYPO3 bereits getestet werden kann. Der Marker `###SUBTEMPLATE###` wird in TYPO3 durch die als subtemplate definierten HTML-Designvorlagen ersetzt.

```

...
<div id="container">
  <div id="header_01">###SERVICENAVIGATION###</div>
  <div id="header_02">
    <div id="flash">
      ###FLASH###
    </div>
    <div id="navi">
      ###NAVIGATION###
    </div>
    <div class="clear"></div>
  </div>
  <div id="line_01"></div>
  <!-- ###DOCUMENT### -->
  <div id="content_wrap">
    <div id="sub_navi">
      ###SUBNAVIGATION###<br />
      ###LEFT###
    </div>
    <div id="img">###IMG###</div>
    <div id="content">
      ###BREADCRUMP###
      ###CONTENT###
    </div>
    <div id="right">###RIGHT###</div>
    <div class="clear"></div>
  </div>
  <!-- ###DOCUMENT### -->
</div>
<div id="line_02"></div>
<div id="footer">###FOOTER###</div>
...

```

Anhang 2 - sub.html - das default Subtemplate. Es zeichnet sich durch ein vierspaltiges Design aus. Wichtig hierbei sind wieder nur die Angaben zwischen den Markern `###DOCUMENT###` - alle anderen Angaben verlieren innerhalb TYPO3s an Bedeutung.

```

...
<!-- ###DOCUMENT### -->
<div id="content_wrap_startseite">
  <div id="content_real_full_size">
    ###CONTENT###
  </div>
  <div id="right">###RIGHT###</div>
  <div class="clear"></div>
</div>
<!-- ###DOCUMENT### -->
...

```

Anhang 3 - sub_startseite.html - eine Alternative zum default Subtemplate. Hier wird ein zweispaltiges Design erzeugt. Die erste große Spalte nimmt den Platz der ersten drei Spalten des default Subtemplates ein.

```

...
<!-- ###DOCUMENT### -->
<div id="content_wrap">
  <div id="sub_navi">
    ###SUBNAVIGATION###<br />
    ###LEFT###
  </div>
  <div id="img">###IMG###</div>
  <div id="content_full_size">
    ###BREADCRUMP###
    ###CONTENT###
  </div>
  <div class="clear"></div>
</div>
<!-- ###DOCUMENT### -->
...

```

Anhang 4 - one_column.html - ein dreispaltiges Design. Der Contentbereich (dritte Spalte) nimmt den Platz der 4. Spalte des default Subtemplates mit ein.

6.2. TypeScript

```

### flash ###
flash < plugin.tx_flashpageheader_pil

### main menu ###
mainmenu = COA
mainmenu {
    10 = HMENU
    10 {
        entryLevel = 0
        1 = TMENU
        1 {
            wrap = <ul>|
            noBlur = 1
            expAll = 0
            NO = 1
            NO {
                allWrap = <li>|</li>
                ATagTitle.field = description //
            subtitle // abstract // title
                stdWrap.htmlSpecialChars = 1
            }
            ACT = 1
            ACT < .NO
            ACT.allWrap = <li class="active">|</li>
        }
    }
}

### sub menu ###
submenu = COA
submenu {
    10 = HMENU
    10 {
        entryLevel = 1
        excludeUidList = 2,8,58,59,62
        1 = TMENU
        1 {
            wrap = <ul>|</ul>
            noBlur = 1
            expAll = 0
            NO = 1
            NO {
                allWrap = <li>|</li>
                ATagTitle.field = description //
            subtitle // abstract // title
                stdWrap.htmlSpecialChars = 1
            }
            ACT = 1

```



```

        ACT < .NO
        ACT.allWrap = <li class="active">|</li>
    }
    2 < .1
    2 {
        wrap = <li><ul>|</ul></li>
        expAll = 0
        NO.allWrap = <li>|</li>
        ACT = 1
        ACT < .NO
        ACT.allWrap = <li class="active">|</li>
    }
}

### headerlink ###
headerlink = TEXT
headerlink.value = .
headerlink.typolink.parameter = 2

### searchbox ###
searchbox < plugin.tx_macinasearchbox_pi1

### service menu ###
servicemenu = COA
servicemenu {
    10 = HMENU
    10 {
        special = directory
        special.value = {$serviceMenu}
        1 = TMENU
        1 {
            noBlur = 1
            wrap = <ul>|</ul>
            NO = 1
            NO {
                linkWrap = <li
class="service_menu_5">|</li> |*| <li
class="service_menu_4">|</li> || <li
class="service_menu_3">|</li> |*| <li
class="service_menu_2">|</li> || <li
class="service_menu_1">|</li>
                ATagTitle.field = description //
abstract // subtitle // title
            }
        }
    }
}

### breadcrumb menu ###
breadcrumb = COA

```

```

breadcrumb {
  10 = HMENU
  10 {
    special = rootline
    special.range = 1 | -1
    stdWrap.required = 1
    includeNotInMenu = 1
    1 = TMENU
    1 {
      #wrap = >&nbsp; |
      NO = 1
      NO {
        # Pfeil in Breadcrumb-Navigation nur
        zwischen zwei Items...
        allWrap = | &nbsp;>&nbsp; |*| |
        &nbsp;>&nbsp; |*| | &nbsp;>&nbsp;
        # nach 25 Zeichen abschneiden
        stdWrap.crop = 35 | ... | 1
      }
      CUR = 1
      CUR {
        doNotLinkIt = 1
        stdWrap.crop = 30 | ... | 1
      }
    }
  }
}
breadcrumb.wrap = <div id="breadcrumb">|</div>

### content left ###
contentleft = COA
contentleft {
  10 < styles.content.getLeft
  10.renderObj {
    stdWrap.wrap = <div
class="contentElement">|</div>
  }
}

contentimg = COA
contentimg {
  10 = IMG_RESOURCE
  10 {
    file {
      import = uploads/media/
      import {
        data = levelmedia:-1, slide
        listNum = 0
      }
    }
    stdWrap.outerWrap.cObject = COA
  }
}

```

```

        stdWrap.outerWrap.cObject {
            10 = TEXT
            10.value = <div id="middle_img"></div>
        }
    }

    }

    ### content main ###
    contentmain < styles.content.get
    contentmain {
        wrap = <!--TYPO3SEARCH_begin-->|<!--
TYPO3SEARCH_end-->
        renderObj {
            stdWrap.wrap = <div
class="contentElement">|</div>
        }
    }

    ### content right ###
    contentright = COA
    contentright {
        10 < fontsize
        20 < printpage
        30 < tipafriend
        40 < styles.content.getRight
        40.renderObj {
            stdWrap.wrap = <div
class="contentElement">|</div>
        }
    }

    ### footer ###
    footer = COA
    footer {
        2.value = flashplayer

    ### template selector (extension rlmp_tmplselector) ###
    plugin.tx_rlmptmplselector_pil {
        templatePathMain =
fileadmin/kkh/template/html/main/
        defaultTemplateNameMain = main.html
        inheritMainTemplates = 0

        templatePathSub = fileadmin/kkh/template/html/sub/
        defaultTemplateNameSub = sub.html
        inheritSubTemplates = 0
    }

    ### sub template ###

```

```

subtemplate = TEMPLATE
subtemplate {
    template < plugin.tx_rlmptmplselector_pi1
    template.templateType = sub
    workOnSubpart = DOCUMENT
    marks {
        LEFT < contentleft
        IMG < contentimg
        BREADCRUMP < breadcrump
        CONTENT < contentmain
        RIGHT < contentright
        SUBNAVIGATION < submenu
    }
}

### page ###
page = PAGE
page {
    meta {
        keywords.field = keywords
        robots = INDEX,FOLLOW
        keywords.ifEmpty (
            Klinikum Schönebeck
        )
    }

    includeCSS {
        file1 = fileadmin/kkh/template/css/kkh.css
    }

    headerData {
        10 = TEXT
        10.value (
            <link rel="shortcut icon"
href="favicon.ico" />
            <meta name="copyright" content="Klinikum
Schönebeck" />
            <meta name="robots" content="index, follow"
/>
            <meta name="language" content="de" />
            <meta name="page-topic" content="" />
            <meta name="revisit-after" content="7 days"
/>
        )
    }

    10 = TEMPLATE
    10 {
        template < plugin.tx_rlmptmplselector_pi1
        template.templateType = main
        workOnSubpart = DOCUMENT
    }
}

```

```
marks {  
    SUBTEMPLATE < subtemplate  
    HEADERLINK < headerlink  
    SEARCHBOX < searchbox  
    SERVICENAVIGATION < servicemenu  
    FLASH < flash  
    NAVIGATION < mainmenu  
    FOOTER < footer  
}  
}  
}
```

Anhang 5 - setup_page.ts - ein Auszug aus der wichtigsten TypoScript Seite. Hier wird definiert, welche Marker durch welche TypoScript Objekte ersetzt werden, und wie diese den Inhalt erzeugen.

7. Quellenverzeichnis

Literatur

- **Andreas Stöckl, Frank Bongers (2006)**
Einstieg in TYPO3 4.0
2., aktualisierte und erweiterte Auflage 2006
- **Beyer, Thomas (2008)**
Relaunch Internet + Intranet
Pflichtenheft

Internet

- **o.V. - PHP**
<http://de.wikipedia.org/wiki/Php>
Stand 03. August 2009
- **o.V. - Mittwald**
<http://www.mittwald.de>
Stand 05. August 2009
- **o.V. - Content-Management-System**
<http://de.wikipedia.org/wiki/Content-Management-System>
Stand 03. August 2009
- **o.V. - Intranet**
<http://de.wikipedia.org/wiki/Intranet>
Stand 03. August 2009
- **o.V. - Corporate Identity**
http://de.wikipedia.org/wiki/Corporate_Identity
Stand 05. August 2009
- **o.V. - Barrierefreiheit**
<http://de.wikipedia.org/wiki/Barrierefreiheit>
Stand 05. August 2009

- **o.V. - Barrierefreies Internet**
http://de.wikipedia.org/wiki/Barrierefreies_Internet
Stand 05. August 2009
- **o.V. - TYPO3**
<http://de.wikipedia.org/wiki/TYPO3>
Stand 06. August 2009
- **o.V. - System Requirements**
<http://TYPO3.org/1275.0.html>
Stand 04. August 2009
- **o.V. - Klinikum Schönebeck: Startseite**
<http://www.kreiskrankenhaus-schoenebeck.de/>
Stand 10. August 2009
- **o.V. - Klinikum Schönebeck: Geschichte**
<http://www.kreiskrankenhaus-schoenebeck.de/unser-krankenhaus/geschichte.html>
Stand 10. August 2009
- **o.V. - mediatack GmbH**
<http://www.mediatack.de>
Stand 18. August 2009
- **o.V. - Media (DAM)**
<http://typo3.org/extensions/repository/view/dam/current/>
Stand 08. August 2009
- **o.V. - Calendar Base**
<http://typo3.org/extensions/repository/view/cal/current/>
Stand 06. August 2009
- **o.V. - Page Template Selector**
http://typo3.org/extensions/repository/view/rImp_tmplselector/current/
Stand 18. August 2009
- **o.V. - TemplaVoila!**
<http://typo3.org/extensions/repository/view/templavoila/current/>
Stand 18. August 2009
- **o.V. - JW Calendar**
http://typo3.org/extensions/repository/view/jw_calendar/current/
Stand 08. August 2009

-
- **o.V. - JW Calendar**
http://typo3.org/documentation/document-library/extension-manuals/jw_calendar/1.3.20/view/1/1/#id3554775
Stand 08. August 2009
 - **o.V. - Indexed Search Documentation and Reference**
http://typo3.org/extensions/repository/view/doc_indexed_search/current/
Stand 10. August 2009
 - **o.V. - TYPO3: Lucene vs. Indexed Search - Ein Performancetest**
<http://www.form4.de/blog-news/blog/TYPO3-lucene-vs-indexed-search-ein-performancetest/>
Stand 16. August 2009
 - **o.V. - T3AK09 – SOLR Suche – Lucene based Search Engine**
<http://TYPO3blogger.de/t3ak09-solr-suche-lucene-based-search-engine/>
Stand 16. August 2009
 - **o.V. - Flash Page Header**
http://typo3.org/extensions/repository/view/flash_pageheader/current/
Stand 18. August 2009
 - **o.V. - powermail**
<http://typo3.org/extensions/repository/view/powermail/current/>
Stand 16. August 2009
 - **o.V. - Tip-A-Friend Plus**
http://typo3.org/extensions/repository/view/tipafriend_plus/current/
Stand 17. August 2009
 - **o.V. - Datensicherung**
<http://de.wikipedia.org/wiki/Backup>
Stand 19. August 2009
 - **o.V. - Sicherheit eines TYPO3 Content Managment Systems**
<http://www.aixwave.com/leistungen-produkte/cms-systeme-typo3/typo3-sicherheit.html>
Stand 19. August 2009
 - **o.V. - SQL-Injection**
<http://de.wikipedia.org/wiki/SQL-Injection>
Stand 19. August 2009

- **Jens Tißler - Interview zur TYPO3-Sicherheit**

<http://t3n.yeebase.com/interview-typo3-sicherheit-denkt-keine-fehler-machen-243531/>

Stand 17. August 2009

- **o.V. - ist TYPO3 sicher**

<http://typo3.fruit-lab.de/typo3-sicherheit.html>

Stand 18. August 2009